

日 本 国 特 許 庁  
PATENT OFFICE  
JAPANESE GOVERNMENT



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日  
Date of Application:

1999年12月16日

出 願 番 号  
Application Number:

平成11年特許願第356820号

願 人  
Applicant(s):

ソニー株式会社

CERTIFIED COPY OF  
PRIORITY DOCUMENT

2000年 9月22日

特許庁長官  
Commissioner,  
Patent Office

及川耕造



出願番号 出願特2000-3076577

【書類名】 特許願

【整理番号】 9900865306

【提出日】 平成11年12月16日

【あて先】 特許庁長官殿

【国際特許分類】 G06T 5/00

【発明者】

    【住所又は居所】 東京都品川区東五反田1丁目14番10号 株式会社ソニー木原研究所内

    【氏名】 大木 光晴

【発明者】

    【住所又は居所】 東京都品川区東五反田1丁目14番10号 株式会社ソニー木原研究所内

    【氏名】 浅利 直介

【発明者】

    【住所又は居所】 東京都品川区東五反田1丁目14番10号 株式会社ソニー木原研究所内

    【氏名】 亀田 健司

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

    【代表者】 出井 伸之

【代理人】

    【識別番号】 100082131

    【弁理士】

    【氏名又は名称】 稲本 義雄

    【電話番号】 03-3369-6479

【手数料の表示】

    【予納台帳番号】 032089

    【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9708842

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 境界線検出装置および方法、画像処理装置および方法、並びに記録媒体

【特許請求の範囲】

【請求項 1】 入力された複数の画素の保存を制御する保存制御手段と、  
前記保存制御手段により保存が制御された前記画素の画素値の差分を算出する  
算出手段と、

前記算出手段により算出された前記画素値の差分から、境界の有無を検出する  
検出手段と、

前記算出手段により算出された前記画素値の差分、および前記検出手段により  
検出された境界の有無の情報を基に、境界線の情報を作成する作成手段と  
を備えることを特徴とする境界線検出装置。

【請求項 2】 前記検出手段は、所定の閾値と、前記算出手段により算出さ  
れた前記画素値の差分を比較することにより、境界の有無を検出する  
ことを特徴とする請求項 1 に記載の境界線検出装置。

【請求項 3】 前記作成手段により作成される前記境界線の情報は、前記境  
界線の長さ、前記境界線の方向、および前記境界線の始点と終点を含む  
ことを特徴とする請求項 1 に記載の境界線検出装置。

【請求項 4】 入力された複数の画素の保存を制御する保存制御ステップと  
前記保存制御ステップの処理により保存が制御された前記画素の画素値の差分  
を算出する算出ステップと、  
前記算出ステップの処理により算出された前記画素値の差分から、境界の有無  
を検出する検出ステップと、  
前記算出ステップの処理により算出された前記画素値の差分、および前記検出  
ステップの処理により検出された境界の有無の情報を基に、境界線の情報を作成  
する作成ステップと  
を含むことを特徴とする境界線検出方法。

【請求項 5】 入力された複数の画素の保存を制御する保存制御ステップと

前記保存制御ステップの処理により保存が制御された前記画素の画素値の差分を算出する算出ステップと、

前記算出ステップの処理により算出された前記画素値の差分から、境界の有無を検出する検出ステップと、

前記算出ステップの処理により算出された前記画素値の差分、および前記検出ステップの処理により検出された境界の有無の情報を基に、境界線の情報を作成する作成ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項 6】 入力された複数の画素の保存を制御する保存制御手段と、

前記保存制御手段により保存が制御された前記画素に隣接する境界線の有無を検出する検出手段と、

前記保存制御手段により保存が制御された前記画素に対する境界線の位置を算出する位置算出手段と、

前記位置算出手段により算出された位置に従って、前記画素の画素値に対して重み付けを行う重み付け手段と、

前記重み付け手段により重み付けが施された前記画素を出力する出力手段とを備えることを特徴とする画像処理装置。

【請求項 7】 入力された複数の画素の保存を制御する保存制御ステップと

前記保存制御ステップの処理により保存が制御された前記画素に隣接する境界線の有無を検出する検出ステップと、

前記保存制御ステップの処理により保存が制御された前記画素に対する境界線の位置を算出する位置算出ステップと、

前記位置算出ステップの処理により算出された位置に従って、前記画素の画素値に対して重み付けを行う重み付けステップと、

前記重み付けステップの処理により重み付けが施された前記画素を出力する出力ステップと

を含むことを特徴とする画像処理方法。

【請求項 8】 入力された複数の画素の保存を制御する保存制御ステップと

前記保存制御ステップの処理により保存が制御された前記画素に隣接する境界線の有無を検出する検出ステップと、

前記保存制御ステップの処理により保存が制御された前記画素に対する境界線の位置を算出する位置算出ステップと、

前記位置算出ステップの処理により算出された位置に従って、前記画素の画素値に対して重み付けを行う重み付けステップと、

前記重み付けステップの処理により重み付けが施された前記画素を出力する出力ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、境界線検出装置および方法、画像処理装置および方法、並びに記録媒体に関し、注目画素と、それに隣接する画素との画素値の差分を比較することによって境界線を検出し、注目画素と境界線との位置関係に基づいて、画素値に重み付けを施すことにより、サンプルレートをあげることなく、簡単な構成でジャギーを抑えることができる境界線検出装置および方法、画像処理装置および方法、並びに記録媒体に関する。

【0002】

【従来の技術】

例えば、コンピュータ等を用いたゲームにおいて、その表示画面に表示される画像に対応する画像データを求めるために、コンピュータにより、通常、浮動小数点演算を用いて高精度な演算が実行されている。そして、表示画面は、画素（ピクセル）が 2 次元に並べられることにより表現される。すなわち、画素値（色データ）が指定された画素の集合により画像データが表示される。

## 【0003】

例えば、コンピュータにより計算された画像データが、図1に表されるような、斜めの境界線1によって分けられた白の部分2と黒の部分3からなる画像であった場合、これを表示するための各画素は、その位置に従った画像データの画素値を持つため、白の部分2に含まれる画素は白の色データ、黒の部分3に含まれる画素は黒の色データを持つ。すなわち、各画素は、図2に示されるような画素値に指定される。

## 【0004】

図3は、図2に示される画素により表示される表示画面を示す。このように、基となる画像データの境界線は、斜めの直線であるが、表示される画面では、白と黒の境界が階段状に表示されてしまう。このように、本来直線であるべき部分が、階段状に表示されてしまう現象をジャギーという。

## 【0005】

これを解決するためには、図4に示すように、境界線1の付近の画素を、境界をはさんだ2つの画素値が表す色データの間色（この場合は、グレー）で表示するようにすればよい。すなわち、表示画像の画素を求める場合、理想的には、図5に示されるように、コンピュータで演算された高精度の画像データにおいて、注目する画素（I，J）を中心とする、一辺が表示画像の画素間の距離 $\alpha$ と等しい正方形4の内部の画素値の平均値を求めて、その値を表示画像の画素値とするとよい。そのための手法として、スーパーサンプリングと呼ばれる画像処理方法がある。

## 【0006】

スーパーサンプリングとは、画面上のある1画素に対するデータを求めるときに、画素をさらに小さなサブピクセルという単位に分割して、それらの平均値をもってピクセルデータとする方法である。例えば1画素を縦横4分割して（1/16分に分割して）それぞれの仮想的な画素に対してレンダリングを行い、最終的にはそれらの平均値をその画素の画素値とする。すなわち、サンプリングレートをあげることにより、ジャギーを抑える。

## 【0007】

【発明が解決しようとする課題】

しかしながら、単純にスーパーサンプリングを実行すると、レンダリング時間がサブピクセルの数に比例して増加してしまう。従って、画像処理にかかる時間が大幅に増加してしまうため、例えば、ゲームなどのように、画像処理速度を落とせない用途においては、演算部や記憶部に、高性能なものを使用する必要があり、大幅なコストアップにつながっていた。

【0008】

本発明はこのような状況に鑑みてなされたものであり、サンプリングレートを増やすことなく、境界線を検出することにより、より簡単な構成でジャギーを除去するものである。

【0009】

【課題を解決するための手段】

請求項1に記載の境界線検出装置は、入力された複数の画素の保存を制御する保存制御手段と、保存制御手段により保存が制御された画素の画素値の差分を算出する算出手段と、算出手段により算出された画素値の差分から、境界の有無を検出する検出手段と、算出手段により算出された画素値の差分、および検出手段により検出された境界の有無の情報を基に、境界線の情報を作成する作成手段とを備えることを特徴とする。

【0010】

請求項4に記載の境界線検出方法は、入力された複数の画素の保存を制御する保存制御ステップと、保存制御ステップの処理により保存が制御された画素の画素値の差分を算出する算出ステップと、算出ステップの処理により算出された画素値の差分から、境界の有無を検出する検出ステップと、算出ステップの処理により算出された画素値の差分、および検出ステップの処理により検出された境界の有無の情報を基に、境界線の情報を作成する作成ステップとを含むことを特徴とする。

【0011】

請求項5に記載の記録媒体に記録されるプログラムは、入力された複数の画素の保存を制御する保存制御ステップと、保存制御ステップの処理により保存が制



御された画素の画素値の差分を算出する算出ステップと、算出ステップの処理により算出された画素値の差分から、境界の有無を検出する検出ステップと、算出ステップの処理により算出された画素値の差分、および検出ステップの処理により検出された境界の有無の情報を基に、境界線の情報を作成する作成ステップとを含むことを特徴とする。

## 【 0 0 1 2 】

請求項 6 に記載の画像処理装置は、入力された複数の画素の保存を制御する保存制御手段と、保存制御手段により保存が制御された画素に隣接する境界線の有無を検出する検出手段と、保存制御手段により保存が制御された画素に対する境界線の位置を算出する位置算出手段と、位置算出手段により算出された位置に従って、画素の画素値に対して重み付けを行う重み付け手段と、重み付け手段により重み付けが施された画素を出力する出力手段とを備えることを特徴とする。

## 【 0 0 1 3 】

請求項 7 に記載の画像処理方法は、入力された複数の画素の保存を制御する保存制御ステップと、保存制御ステップの処理により保存が制御された画素に隣接する境界線の有無を検出する検出ステップと、保存制御ステップの処理により保存が制御された画素に対する境界線の位置を算出する位置算出ステップと、位置算出ステップの処理により算出された位置に従って、画素の画素値に対して重み付けを行う重み付けステップと、重み付けステップの処理により重み付けが施された画素を出力する出力ステップとを含むことを特徴とする。

## 【 0 0 1 4 】

請求項 8 に記載の記録媒体に記録されるプログラムは、入力された複数の画素の保存を制御する保存制御ステップと、保存制御ステップの処理により保存が制御された画素に隣接する境界線の有無を検出する検出ステップと、保存制御ステップの処理により保存が制御された画素に対する境界線の位置を算出する位置算出ステップと、位置算出ステップの処理により算出された位置に従って、画素の画素値に対して重み付けを行う重み付けステップと、重み付けステップの処理により重み付けが施された画素を出力する出力ステップとを含むことを特徴とする。

## 【 0 0 1 5 】

請求項 1 に記載の境界線検出装置、請求項 4 に記載の境界線検出方法、請求項 5 に記載の記録媒体においては、入力された複数の画素が保存され、保存されている画素の画素値の差分が算出され、画素値の差分から、境界の有無が検出され、画素値の差分、および境界の有無の情報を基に、境界線の情報が作成される。

## 【 0 0 1 6 】

請求項 6 に記載の画像処理装置、請求項 7 に記載の画像処理方法、請求項 8 に記載の記録媒体においては、入力された複数の画素が保存され、画素に隣接する境界線の有無が検出され、画素に対する境界線の位置が算出され、算出された位置に従って、画素の画素値に対して重み付けが行われ、重み付けが施された画素が出力される。

## 【 0 0 1 7 】

## 【発明の実施の形態】

図6は、画像処理装置の構成を示すブロック図である。

## 【 0 0 1 8 】

制御部 1 1 は、ユーザが、例えば、キーボード、マウス、ジョイスティック、各種ボタンなどからなる入力部 1 2 を用いて指令を入力すると、入出力インターフェース 1 3 および内部バス 1 4 を介してその指令を表す信号の入力を受ける。そして、制御部 1 1 は、入力される信号に従って、例えば、ROM (Read Only Memory)、もしくはRAM (Random Access Memory) からなるメモリ 1 5 に格納されているプログラムを実行する。また、制御部 1 1 は、通信部 1 6 および内部バス 1 4 を介して、インターネット等のネットワークを介して入力されるプログラムや各種データをメモリ 1 5 に保存する。更に、制御部 1 1 は、その処理結果を、例えば、内部バス 1 4、入出力インターフェース 1 3 および出力部 1 7 を介して、LCD (Liquid Crystal Display) などよりなるモニタ 1 8 に、必要に応じて出力し、表示させたり、音声データである場合は、内部バス 1 4、入出力インターフェース 1 3 および出力部 1 7 を介してスピーカ 1 9 に出力して、音声を出力させる。

## 【 0 0 1 9 】

メモリ 1 5 には、インストールされたプログラムのほかに、例えば、制御部 1 1 の処理によって生成されたデータなどが保存されている。また、入出力インターフェース 1 3 には、ドライブ 2 0 も接続されており、磁気ディスク 2 1、光ディスク 2 2、光磁気ディスク 2 3、および半導体メモリ 2 4 などとデータの授受を行う。

#### 【0 0 2 0】

画像処理部 2 5 は、内部バス 1 4 に接続され、制御部 1 1 から入力される制御信号に従って、後述する処理を実行することにより、入力された画像データをモニタ 1 8 の表示に適応するデータに変換する。

#### 【0 0 2 1】

図 7 は、画像処理部 2 5 の詳細な構成を示すブロック図である。

#### 【0 0 2 2】

画像データは、通常、ラスタスキャン順に入力される。すなわち、入力される画像データの各画素の座標を  $(I, J)$  で表すと、画像データの画素数が、横  $W$ 、縦  $H$  の  $W \times H$  で表される場合、 $(0, 0)$   $(0, 1)$   $(0, 2) \cdots (0, W-1)$   $(1, 0)$   $(1, 1)$   $(1, 2) \cdots (1, W-1) \cdots (H-1, 0)$   $(H-1, 1) \cdots (H-1, W-1)$  の順に入力される。

#### 【0 0 2 3】

入力された画素は、メモリ 3 1 に一時格納され、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、メモリ 3 1 から必要な画素の画素値を読み込む。横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は境界線検出用と、境界線特定用に、入力された画素の座標を表すためのレジスタをそれぞれ有し、横方向の座標の値を  $J$ 、縦方向の座標の値を  $I$  とする。

#### 【0 0 2 4】

横方向境界線検出部 3 2 は、制御部 1 1 から入力される制御信号に従って、境界線の検出に必要な画素をメモリ 3 1 から入力され、図 9 を用いて後述する処理により、縦に並んだ 2 つの画素間の画素値の差の絶対値を求め、図 1 0 を用いて後述する処理により、横方向の境界線を検出し、その結果を横方向境界情報メモリ 3 3 に保存する。境界線の検出には、注目画素の周辺の画素値の差の絶対値も

必要であるため、横方向境界線検出部 3 2 は、内部に図示しないメモリを有し、複数の画素値の差の絶対値の算出結果を一時保存しておくことができる。

#### 【 0 0 2 5 】

横方向境界線メモリに保存されるデータは、横方向の境界線の長さを示すlength、境界線のその行における開始位置を示すstart (X, Y) ((X, Y) は開始位置の座標を示す)、境界線が、他の行のどちらの方向から延びてきたのかを示すFrom (1 の場合左下、-1 の場合左上、0 の場合この行が始点であることを示す)、および、境界線が他の行に延びていく方向を示すTo (1 の場合右下、-1 の場合右上、0 の場合この行が終端であることを示す) からなる境界線情報、および、最新の処理 (一つ前の注目画素に対する処理) の結果、境界が存在したか否かを示すEdgestate (0 の場合境界なし、1 の場合境界ありを示す) である。

#### 【 0 0 2 6 】

すなわち、図 8 に示されるように、ある境界線検出行に着目して境界線が検出され、例えば、境界線 A は、"Start (A, B), From=-1, Length=3, To=0" という境界線情報で表され、境界線 B は、"Start (A, B), From=0, Length=2, To=-1" という境界線情報で表される。

#### 【 0 0 2 7 】

また、縦方向境界線検出部 3 4 は、制御部 1 1 から入力される制御信号に従って、境界線の検出に必要な画素をメモリ 3 1 から入力され、図 9 を用いて後述する処理により、横に並んだ 2 つの画素間の画素値の差の絶対値を求め、図 1 7 を用いて後述する処理により、縦方向の境界線を検出し、その結果を縦方向境界情報メモリ 3 5 に保存する。縦方向境界線検出部 3 4 も、内部に図示しないメモリを有し、複数の画素値の差の絶対値の算出結果を一時保存しておくことができる。

#### 【 0 0 2 8 】

縦方向境界線メモリに保存されるデータは、縦方向の境界線の長さを示すPrevlength、境界線のこの列における開始位置を示すstart (X, Y)、境界線が他の列のどちらの方向から延びてきたのかを示すFrom、および、境界線が他の列に延びていく方向を示すToからなる境界線情報、最新の処理 (一つ前の注目画素に対

する処理)の結果、境界が存在したか否かを示すEdgestate、および、境界が存在しない座標を表すNoedge= (I, J) である。

#### 【 0 0 2 9 】

画素値変換部 3 6 は、その内部に、注目画素と境界線との距離検出処理用と、画素値変換処理用に、入力された画素の座標を表すためのレジスタをそれぞれ有し、距離検出用レジスタは、横方向の座標の値を B、縦方向の座標の値を A とし、画素値変換用レジスタは、横方向の座標の値を J、縦方向の座標の値を I とする。そして、画素値変換部 3 6 は、横方向境界線情報メモリ 3 3 および縦方向境界線情報メモリ 3 5 から、境界線情報を読み込み、かつ、メモリ 3 1 より入力された画素を読み込み、図 2 1 乃至 2 3 を用いて後述する処理により、出力すべき画素値を演算し、演算結果に従って画素値を変換し、出力する。

#### 【 0 0 3 0 】

次に、図 9 のフローチャートを参照して、境界線検出処理について説明する。ここでは、1 フレーム分の画像データの処理について説明するが、動画像データの処理を実行する場合、複数のフレームを連続して処理することにより、動画像データの処理を行うことができる。

#### 【 0 0 3 1 】

ステップ S 1 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、画素の縦方向座標を表すための境界線検出用レジスタの値を  $I = 0$  とする。ステップ S 2 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの値 I を参照し、 $I = H - 1$  であるか否か（すなわち、最後から 2 番目の行までの画素を注目画素とした処理が終了したか否か）を判断する。

#### 【 0 0 3 2 】

ステップ S 2 において、 $I = H - 1$  であると判断された場合、処理が終了される。ステップ S 2 において、 $I = H - 1$  ではないと判断された場合、ステップ S 3 において、縦方向境界線検出部 3 4 および横方向境界線検出部 3 2 は、画素の横方向座標を表すためのレジスタの値を  $J = 0$  とする。

#### 【 0 0 3 3 】

ステップ S 4 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの値  $J$  を参照し、 $J=W-1$  か否か（すなわち、その行の最後から 2 番目の画素を注目画素とした処理が終了したか否か）を判断する。ステップ S 4 において、 $J=W-1$  であると判断された場合、ステップ S 5 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの  $I$  の値を  $I+1$  とする。その後処理はステップ S 2 に戻り、それ以降の処理が繰り返される。

## 【 0 0 3 4 】

ステップ S 4 において、 $J=W-1$  ではないと判断された場合、ステップ S 6 において、横方向境界線検出部 3 2 は、メモリ 3 1 に保存されている画像データから、画素  $(I, J)$  および画素  $(I+1, J)$ （すなわち、注目画素と、その下の画素）を読み込み、その 2 つの画素の画素値の差の絶対値を求め、内部のメモリに保存する。

## 【 0 0 3 5 】

ステップ S 7 において、縦方向境界線検出部 3 4 は、画素  $(I, J)$  および画素  $(I, J+1)$ （すなわち、注目画素と、その右の画素）を読み込み、その 2 つの画素の画素値の差の絶対値を求め、内部のメモリに保存する。

## 【 0 0 3 6 】

ステップ S 8 において、画素値変換部 3 6 は、画素  $(I, J)$  を読み込み、内部のメモリに保存する。

## 【 0 0 3 7 】

ステップ S 9 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの  $J$  の値を  $J+1$  とする。その後処理はステップ S 4 に戻り、それ以降の処理が繰り返される。

## 【 0 0 3 8 】

なお、境界線検出処理において、メモリ 3 1 は、1 フレーム分の画像データの画素値を保存した状態で、この処理を実行してもよいし、縦方向境界線検出部 3 4、横方向境界線検出部 3 2、および画素値変換部 3 6 に注目画素およびその周辺の画素の画素値を出力しつつ、新たな画素の入力を受けるようにしてもよい。

## 【 0 0 3 9 】

次に、図 1 0 および図 1 1 のフローチャートを参照して、横境界検出処理について説明する。

## 【 0 0 4 0 】

ステップ S 2 1 において、横方向境界線検出部 3 2 は、画素の縦方向座標を表すための境界線特定用レジスタの値を  $I=0$  とする。ステップ S 2 2 において、横方向境界線検出部 3 2 は、境界線特定用レジスタの値  $I$  を参照し、 $I=H-1$  であるか否か（すなわち、最後から 2 番目の行までの画素を注目画素とした処理が終了したか否か）を判断する。

## 【 0 0 4 1 】

ステップ S 2 2 において、 $I=H-1$  であると判断された場合、処理が終了される。ステップ S 2 2 において、 $I=H-1$  ではないと判断された場合、ステップ S 2 3 において、横方向境界線検出部 3 2 は、Edgestate の値を 0 とし、境界線特定用レジスタの値を  $J=0$  とする。

## 【 0 0 4 2 】

ステップ S 2 4 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの値  $J$  を参照し、 $J=W$  か否か（すなわち、その行の最後の画素を注目画素とした処理が終了したか否か）を判断する。ステップ S 2 4 において、 $J=W$  であると判断された場合、ステップ S 2 5 において、横方向境界線検出部 3 2 は、Edgestate=1 であるか否か（すなわち、その行の最後の画素に境界があるか否か）を判断する。

## 【 0 0 4 3 】

ステップ S 2 5 において Edgestate=1 ではないと判断された場合、処理はステップ S 2 8 に進む。ステップ S 2 5 において Edgestate=1 であると判断された場合、画像の端に境界線が届いている、すなわち、この境界線は、この行に終端を有していることになるので、縦方向境界線検出部 3 4 および横方向境界線検出部 3 2 は、ステップ S 2 6 において、 $To=0$  とし、ステップ S 2 7 において、縦方向境界線情報メモリ 3 5 および横方向境界線情報メモリ 3 3 に Start (X, Y) 、 From、To などの境界線情報を書き込む。

## 【 0 0 4 4 】

ステップ S 2 8 において、横方向境界線検出部 3 2 および縦方向境界線検出部 3 4 は、境界線検出用レジスタの I の値を I + 1 とする。その後処理はステップ S 2 2 に戻り、それ以降の処理が繰り返される。

## 【 0 0 4 5 】

ステップ S 2 4 において、 $J=W$ ではないと判断された場合、ステップ S 2 9 において、横方向境界線検出部 3 2 は、内部メモリに保存されている各画素の差の絶対値の算出結果から、画素 (I, J) と画素 (I + 1, J) の間（すなわち、図 1 2 において a で示される部分）に境界があるか否かを判断する。例えば、比較する 2 つの画素値の差の絶対値に閾値を設け、画素値の差の絶対値と閾値とを比較して、閾値以上の場合は境界あり、閾値以下の場合は境界なしと判断するようにしてもよいし、その他の判断方法を用いてもよい。

## 【 0 0 4 6 】

ステップ S 2 9 において、境界がないと判断された場合、処理は、図 1 1 のステップ S 5 1 に進む。ステップ S 2 9 において、境界があると判断された場合、ステップ S 3 0 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J) と画素 (I + 1, J) との画素値の差の絶対値と、画素 (I - 1, J - 1) と画素 (I, J - 1) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。ここで、2箇所において境界の特徴が類似しているということは、同一の境界線を有しているということなので、ステップ S 3 0 は、図 1 2 において b で示される境界線が存在するか否かを判断していることにほかならない。

## 【 0 0 4 7 】

ステップ S 3 0 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、処理は、ステップ S 3 4 に進む。ステップ S 3 0 において、それぞれの画素間の境界の特徴が類似していると判断された場合、ステップ S 3 1 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている Edgestate の値が 1 か否か（すなわち、直前の注目画素に対する処理において、図 1 2 の c で示されるエリアに境界線が存在したと判断されたか否か）を判断す



る。

【 0 0 4 8 】

ステップ S 3 1 において、Edgestate の値が 1 ではないと判断された場合、処理は、ステップ S 3 3 に進む。ステップ S 3 1 において、Edgestate の値が 1 であると判断された場合、図 1 2 の c に存在する境界線 d は、c と a の間に終端を有しているので、ステップ S 3 2 において、横方向境界線検出部 3 2 は、c に存在する境界線 d の境界線情報を、To=0 (すなわち、終端であることを示す) とし、境界線 d を示す境界線情報 "Start, From, Length, To" を、ひとつのデータの組として、横方向境界線情報メモリ 3 3 に書き込む。

【 0 0 4 9 】

ステップ S 3 3 において、横方向境界線検出部 3 2 は、ステップ S 2 9 において検出された境界線の情報として、"Start (I, J), From=-1, Length=1" を図示しない内部メモリに記録する。この画素は、境界線が始まった点であり、まだ、その方向はわからないため、To の値はここでは設定されない。そして処理は、ステップ S 4 3 に進む。

【 0 0 5 0 】

ステップ S 3 1 において、比較された境界の特徴が類似していないと判断された場合、ステップ S 3 4 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J) と画素 (I+1, J) との画素値の差の絶対値と、画素 (I, J-1) と画素 (I+1, J-1) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。これは、図 1 3 において、e で示される境界線が存在するか否かを判断していることにほかならない。

【 0 0 5 1 】

ステップ S 3 4 において、それぞれの画素間の境界の特徴が類似していると判断された場合、ステップ S 3 5 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている境界線情報 "Length" の値を、"Length=Length+1" とする。そして処理は、ステップ S 4 3 に進む。

【 0 0 5 2 】

ステップ S 3 4 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、ステップ S 3 6 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J) と画素 (I + 1, J) との画素値の差の絶対値と、画素 (I + 1, J - 1) と画素 (I + 2, J - 1) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。これは、図 1 4 において f で示される境界線が存在するか否かを判断していることにほかならない。

## 【0 0 5 3】

ステップ S 3 6 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、処理は、ステップ S 4 0 に進む。ステップ S 3 6 において、それぞれの画素間の境界の特徴が類似していると判断された場合、ステップ S 3 7 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている Edgestate の値が 1 か否か（すなわち、直前の注目画素に対する処理において、図 1 4 の c で示されるエリアに境界線が存在したと判断されたか否か）を判断する。

## 【0 0 5 4】

ステップ S 3 7 において、Edgestate の値が 1 ではないと判断された場合、処理は、ステップ S 3 9 に進む。ステップ S 3 7 において、Edgestate の値が 1 であると判断された場合、図 1 4 の c に存在する境界線 g は、c と a の間に終端を有しているので、ステップ S 3 8 において、横方向境界線検出部 3 2 は、c に存在する境界線 g の境界線情報を、To = 0（すなわち、終端であることを示す）とし、境界線 g を示す境界線情報 "Start, From, Length, To" を、ひとつのデータの組として、横方向境界線情報メモリ 3 3 に書き込む。

## 【0 0 5 5】

ステップ S 3 9 において、横方向境界線検出部 3 2 は、ステップ S 2 9 において検出された境界線の情報として、"Start (I, J), From = 1, Length = 1" を図示しない内部メモリに記録する。この画素は、境界線が始まった点であり、まだ、その方向はわからないため、To の値はここでは設定されない。そして処理は、ステップ S 4 3 に進む。

## 【 0 0 5 6 】

ステップ S 3 6 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、ステップ S 2 9 において検出された境界線は、注目画素の左に存在する 3 つの画素のいずれにも類似していないのであるから、この画素から、新たな境界線が発生したと判断される。そして、ステップ S 4 0 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている Edgestate の値が 1 か否か（すなわち、直前の注目画素に対する処理において、図 1 5 の c で示されるエリアに境界線が存在したと判断されたか否か）を判断する。

## 【 0 0 5 7 】

ステップ S 4 0 において、Edgestate の値が 1 ではないと判断された場合、処理は、ステップ S 4 2 に進む。ステップ S 4 0 において、Edgestate の値が 1 であると判断された場合、図 1 5 の c には境界線 i が存在するが、ステップ S 3 4 において、境界の特徴が類似していないと判断されているので、ここで検出している境界線 h とは別の境界線であることがわかる。従って、ステップ S 4 1 において、横方向境界線検出部 3 2 は、c に存在する境界線 i の境界線情報を、To = 0（すなわち、終端であることを示す）とし、境界線 i を示す境界線情報 "Start, From, Length, To" を、ひとつのデータの組として、横方向境界線情報メモリ 3 3 に書き込む。

## 【 0 0 5 8 】

ステップ S 4 2 において、横方向境界線検出部 3 2 は、ステップ S 2 9 において検出された境界線の情報として、"Start (I, J), From = 0, Length = 1" を図示しない内部メモリに記録する。この画素は、境界線が始まった点であり、まだ、その方向はわからないため、To の値はここでは設定されない。

## 【 0 0 5 9 】

ステップ S 4 3 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている Edgestate の値を、"Edgestate = 1" とする。

## 【 0 0 6 0 】

ステップ S 4 4 において、横方向境界線検出部 3 2 は、境界線検出用レジスタの J の値を J + 1 とする。そして、処理はステップ S 2 4 に戻り、それ以降の処

理が繰り返される。

【0 0 6 1】

次に、図 1 1 のステップ S 5 1 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている Edgestate の値が 1 か否か（すなわち、直前の注目画素に対する処理において、図 1 6 の c で示されるエリアに境界線が存在したと判断されたか否か）を判断する。

【0 0 6 2】

ステップ S 5 1 において、Edgestate の値が 1 ではないと判断された場合、処理は、図 1 0 のステップ S 4 3 に戻り、それ以降の処理が繰り返される。ステップ S 5 1 において、Edgestate の値が 1 であると判断された場合、ステップ S 5 2 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J - 1) と画素 (I + 1, J - 1) との画素値の差の絶対値と、画素 (I - 1, J) と画素 (I, J) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。これは、図 1 6 において 1 で示される境界線が存在するか否かを判断していることにほかならない。

【0 0 6 3】

ステップ S 5 2 において、それぞれの画素間の境界の特徴が類似していると判断された場合、ステップ S 5 3 において、横方向境界線検出部 3 2 は、To = - 1 とし、処理は、ステップ S 5 7 に進む。

【0 0 6 4】

ステップ S 5 2 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、ステップ S 5 4 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J - 1) と画素 (I + 1, J - 1) との画素値の差の絶対値と、画素 (I + 1, J) と画素 (I + 2, J) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。これは、図 1 6 において m で示される境界線が存在するか否かを判断していることにほかならない。

【0 0 6 5】

ステップ S 5 4 において、それぞれの画素間の境界の特徴が類似していると判断された場合、ステップ S 5 5 において、横方向境界線検出部 3 2 は、 $T_o = 1$  とし、処理は、ステップ S 5 7 に進む。

【 0 0 6 6 】

ステップ S 5 2 において、それぞれの画素間の境界の特徴が類似していないと判断された場合、画素 (I, J - 1) と画素 (I + 1, J - 1) との間に存在する境界線は、ここで終端を有するため、ステップ S 5 6 において、横方向境界線検出部 3 2 は、 $T_o$  の値を  $T_o = 0$  とする。

【 0 0 6 7 】

横方向境界線検出部 3 2 は、ステップ S 5 7 において、境界線情報 "Start, From, Length,  $T_o$ " を、ひとつのデータの組として、横方向境界線情報メモリ 3 3 に書き込み、ステップ S 5 8 において、境界線情報を  $Edgestate = 0$  とし、処理は、図 1 0 のステップ S 4 3 に戻り、それ以降の処理が繰り返される。

【 0 0 6 8 】

次に、図 1 7 のフローチャートを参照して、縦方向境界線特定処理について説明する。縦方向境界線特定処理として、図 1 0 および図 1 1 を用いて説明した横境界線特定処理と同様の処理を実行しようとした場合、メモリ 3 1 に画素が入力されるのは、ラスタスキャン順であるため、例えば、境界線が、一番上のラインから一番下のラインまでつながっていた場合、一番上のラインから一番下のラインまでの画素に対する処理が終了するまで、境界線を特定することができない。従って、縦方向境界線特定処理を、図 1 0 および図 1 1 を用いて説明した横境界線特定処理と同様の処理を実行しようとした場合、1 画面分の全ての画素をメモリ 3 1 に保存しておかなければならず、メモリ容量が増大する。図 1 7 を参照して説明する縦方向特定処理は、メモリ容量を削減するため、すでに処理済のデータを用いて、境界線の推測を行う。

【 0 0 6 9 】

ステップ S 7 1 において、縦方向境界線検出部 3 4 は、画素の縦方向座標を表すための境界線特定用レジスタの値を  $I = 0$  とする。ステップ S 7 2 において、縦方向境界線検出部 3 4 は、境界線特定用レジスタの値  $I$  を参照し、 $I = H$  である

か否か（すなわち、最後の行までの画素を注目画素とした処理が終了したか否か）を判断する。

#### 【0070】

ステップS72において、 $I=H$ であると判断された場合、処理が終了される。  
ステップS72において、 $I \neq H$ ではないと判断された場合、ステップS73において、縦方向境界線検出部34は、境界線情報を $Plevlength[J]=1$  ( $J=0, 1, 2, \dots, W-2$ )、境界線特定用レジスタの値を $J=0$ とする。

#### 【0071】

ステップS74において、縦方向境界線検出部34は、境界線検出用レジスタの値 $J$ を参照し、 $J=W-1$ か否か（すなわち、その行の最後から2番目の画素を注目画素とした処理が終了したか否か）を判断する。ステップS74において、 $J=W-1$ であると判断された場合、ステップS75において、縦方向境界線検出部34は、境界線特定用レジスタの値を $I=I+1$ とする。そして処理は、ステップS72に戻り、それ以降の処理が繰り返される。

#### 【0072】

ステップS74において、 $J=W-1$ ではないと判断された場合、ステップS76において、縦方向境界線検出部34は、内部メモリに保存されている各画素の差の絶対値の算出結果から、画素( $I, J$ )と画素( $I, J+1$ )の間（すなわち、図18においてpで示される部分）に境界があるか否かを判断する。

#### 【0073】

ステップS76において、境界がないと判断された場合、ステップS77において、縦方向境界線検出部34は、縦方向境界線情報メモリ35に、 $Noedge=(I, J)$ を書き込み、処理は、ステップS87に進む。ステップS76において、境界があると判断された場合、ステップS76において、縦方向境界線検出部34は、図示しない内部メモリに保存されている画素( $I, J$ )と画素( $I, J+1$ )との画素値の差の絶対値と、画素( $I-1, J$ )と画素( $I-1, J+1$ )との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。ここで、2箇所において境界の特徴が類似しているということは、同一の境界線を有しているということなので、ステップ

S 7 8 は、図 1 8 において r で示される境界線が存在するか否かを判断していることにほかならない。

【 0 0 7 4 】

ステップ S 7 8 において、それぞれの境界の特徴が類似していると判断された場合、ステップ S 7 9 において、縦方向境界線検出部 3 4 は、境界線情報を  $length[J] = length[J] + 1$  とする。そして処理は、ステップ S 8 7 に進む。

【 0 0 7 5 】

ステップ S 7 8 において、それぞれの境界の特徴が類似していないと判断された場合、ステップ S 8 0 において、縦方向境界線検出部 3 4 は、図示しない内部メモリに保存されている画素 (I, J) と画素 (I, J + 1) との画素値の差の絶対値と、画素 (I - 1, J - 1) 画素 (I - 1, J) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。ここで、ステップ S 8 0 は、図 1 8 において s で示される境界線が存在するか否かを判断していることにほかならない。

【 0 0 7 6 】

ステップ S 8 0 において、それぞれの境界の特徴が類似していると判断された場合、ステップ S 8 1 において、縦方向境界線検出部 3 4 は、境界線情報を  $prevlength[J] = prevlength[J - 1]$ ,  $From = -1$ ,  $Start = (I, J)$ ,  $To = 1$  とする。そして処理は、ステップ S 8 5 に進む。

【 0 0 7 7 】

ステップ S 8 0 において、それぞれの境界の特徴が類似していないと判断された場合、ステップ S 8 2 において、横方向境界線検出部 3 2 は、図示しない内部メモリに保存されている画素 (I, J) と画素 (I, J + 1) との画素値の差の絶対値と、画素 (I - 1, J + 1) と画素 (I - 1, J + 2) との画素値の差の絶対値を比較することにより、それぞれの画素間の境界の特徴が類似しているか否かを判断する。ここで、ステップ S 8 2 は、図 1 8 において t で示される境界線が存在するか否かを判断していることにほかならない。

【 0 0 7 8 】

ステップ S 8 2 において、それぞれの境界の特徴が類似していると判断された

場合、ステップ S 8 3 において、縦方向境界線検出部 3 4 は、境界線情報を  $\text{prevlength}[J] = \text{prevlength}[J+1]$  ,  $\text{From} = 1$  ,  $\text{Start} = (I, J)$  ,  $\text{To} = -1$  とする。そして処理は、ステップ S 8 5 に進む。

#### 【0079】

ステップ S 8 2 において、それぞれの境界の特徴が類似していないと判断された場合、ここから境界線が開始しているということであるので、ステップ S 8 4 において、縦方向境界線検出部 3 4 は、境界線情報を  $\text{prevlength}[J] = 1$  ,  $\text{From} = 0$  ,  $\text{Start} = (I, J)$  ,  $\text{To} = 0$  とする。

#### 【0080】

ここで、境界線は、注目画素とその右隣の画素の間（図 18 の p で示される場所）を通過している、すなわち、行ごとで考えた場合は、ここから境界線が開始されているのであるから、ステップ S 8 5 において、縦方向境界線検出部 3 4 は、Length の値を 1 とする。

#### 【0081】

縦方向境界線検出部 3 4 は、ステップ S 8 6 において、境界線情報 "Start, From, Plevlength[J], To" を、ひとつのデータの組として、縦方向境界線情報メモリ 3 5 に書き込み、ステップ S 8 7 において、境界線特定用レジスタの値を  $J = J + 1$  とする。そして、処理は、ステップ S 7 4 に戻り、それ以降の処理が繰り返される。

#### 【0082】

ここでは、一つ前の境界線が、どれぐらいの長さだけこの列にとどまっていたかを用いて、注目する境界線についての推測を行っている。この推測については、例えば、推測の精度を向上させるために、2 つ前までの境界線情報を用いて、2 次式で近似するようにしてもよい。

#### 【0083】

次に、図 19 のフローチャートを参照して、距離検出処理について説明する。

#### 【0084】

ステップ S 9 1 において、画素値変換部 3 6 は、画素の縦方向座標を表すための距離検出用レジスタの値を  $A = 0$  とする。ステップ S 9 2 において、画素値変



換部 36 は、距離検出用レジスタの値  $A$  を参照し、 $A=H-1$  であるか否か（すなわち、最後から 2 番目の行の画素を注目画素とした処理が終了したか否か）を判断する。

## 【0085】

ステップ S92 において、 $A=H-1$  であると判断された場合、処理が終了される。ステップ S92 において、 $A=H-1$  ではないと判断された場合、ステップ S93 において、画素値変換部 36 は、距離検出用レジスタの値を  $B=0$  とする。

## 【0086】

ステップ S94 において、画素値変換部 36 は、距離検出用レジスタの値  $B$  を参照し、 $B=W-1$  か否か（すなわち、その行の最後から 2 番目の画素を注目画素とした処理が終了したか否か）を判断する。ステップ S94 において、 $B=W-1$  であると判断された場合、ステップ S95 において、画素値変換部 36 は、距離検出用レジスタの値を  $A=A+1$  とする。そして処理は、ステップ S92 に戻り、それ以降の処理が繰り返される。

## 【0087】

ステップ S94 において、 $B=W-1$  ではないと判断された場合、ステップ S96 において、画素値変換部 36 は、横方向境界情報メモリに記憶されている  $start(I, J)$  に、 $I=A$ 、 $J \leq B \leq J+length-1$  を満たすものがあるか否かを判断する。

## 【0088】

ステップ S96 において、 $I=A$ 、 $J \leq B \leq J+length-1$  を満たすものがないと判断された場合、処理はステップ S98 に進む。ステップ S96 において、 $I=A$ 、 $J \leq B \leq J+length-1$  を満たすものがあると判断された場合、ステップ S97 において、画素値変換部 36 は、横方向境界情報メモリ 33 に保存されている境界線情報を参照して、例えば、図 20 に示されるような条件に基づいて、画素と境界線との距離を算出する。

## 【0089】

ステップ S98 において、画素値変換部 36 は、縦方向境界情報メモリに記憶されている  $noedge(I, J)$  に、 $I=A$ 、 $J=B$  を満たすものがあるか否かを判断

する。

【 0 0 9 0 】

ステップ S 9 8 において、 $I=A$ 、 $J=B$  を満たすものがあると判断された場合、処理は、ステップ S 1 0 1 に進む。ステップ S 9 8 において、 $I=A$ 、 $J=B$  を満たすものがないと判断された場合、ステップ S 9 9 において、画素値変換部 3 6 は、縦方向境界情報メモリに記憶されている  $\text{start}(I, J)$  に、 $J=B$ 、 $J \leq A \leq I + \text{length} - 1$  を満たすものがあるか否かを判断する。

【 0 0 9 1 】

ステップ S 9 9 において、 $J=B$ 、 $J \leq A \leq I + \text{length} - 1$  を満たすものがないと判断された場合、処理は、ステップ S 1 0 1 に進む。ステップ S 9 9 において、 $J=B$ 、 $J \leq A \leq I + \text{length} - 1$  を満たすものがあると判断された場合、ステップ S 1 0 0 において、画素値変換部 3 6 は、縦方向境界情報メモリ 3 5 に保存されている境界線情報を参照して、例えば、図 2 0 において、B を A に変更し、かつ、J を I に変更した条件を用いることにより、画素と境界線の距離を算出する。ただし、縦方向の境界線特定処理は、図 1 7 を用いて説明したように、推測を用いているため、距離の算出結果が 0 未満の場合は距離を 0 とし、距離の算出結果が 1 以上の場合は距離を 1 としなければならない。

【 0 0 9 2 】

ステップ S 1 0 1 において、画素値変換部 3 6 は、距離算出用レジスタの値を  $B=B+1$  とする。そして処理は、ステップ S 9 4 に進み、それ以降の処理が繰り返される。

【 0 0 9 3 】

次に、図 2 1 のフローチャートを参照して、画素値変換処理について説明する。

【 0 0 9 4 】

ステップ S 1 1 1 において、画素値変換部 3 6 は、画素の縦方向座標を表すための画素値変換用レジスタの値を  $I=0$  とする。ステップ S 1 1 2 において、画素値変換部 3 6 は、画素値変換用レジスタの値 I を参照し、 $I=H-1$  であるか否か（すなわち、最後から 2 番目の行の画素を注目画素とした処理が終了したか否

か)を判断する。

【0095】

ステップS112において、 $I=H-1$ であると判断された場合、処理が終了される。ステップS112において、 $I=H-1$ ではないと判断された場合、ステップS113において、画素値変換部36は、画素値変換用レジスタの値を $J=0$ とする。

【0096】

ステップS114において、画素値変換部36は、画素値変換用レジスタの値 $J$ を参照し、 $J=W-1$ か否か（すなわち、その行の最後から2番目の画素を注目画素とした処理が終了したか否か）を判断する。ステップS114において、 $J=W-1$ であると判断された場合、ステップS115において、画素値変換部36は、画素値変換用レジスタの値を $I=I+1$ とする。そして処理は、ステップS112に戻り、それ以降の処理が繰り返される。

【0097】

ステップS114において、 $I=W-1$ ではないと判断された場合、ステップS116において、画素値変換部36は、横方向境界線情報メモリ33および縦方向境界線情報メモリ35に記憶されている境界線情報を参照して、注目画素（ $I, J$ ）の周辺に境界線があるか否かを判断する。

【0098】

ステップS116において、周辺に境界線がないと判断された場合、処理は、ステップS118に進む。ステップS116において、周辺に境界線があると判断された場合、ステップS117において、画素値変換部36は、横方向境界線情報メモリ33および縦方向境界線情報メモリ35に記憶されている境界線情報を参照して、左右の少なくとも片側に境界線があり、かつ、上下の少なくとも片側に境界線があるか否かを判断する。

【0099】

ステップS116において、周辺に境界線がないと判断された場合、および、ステップS117において、左右の少なくとも片側に境界線があり、かつ、上下の少なくとも片側に境界線があると判断された場合は、その注目画素の周辺に境

界線がないか、あっても、複雑に入り組んだ部分であり、そのままの画素値で注目画素を出力しても、出力画像に何ら影響を与えないため、ステップS118において、画素値変換部36は、図9のステップS8において読み込んだ画素(I、J)をそのまま出力する。

【0100】

ステップS117において、左右のどちらにも境界線がない、もしくは上下のどちらにも境界線がないと判断された場合は、ステップS119において、画素値変換部36は、左右の少なくとも片側に境界線があり、かつ、上下のいずれにも境界線がないか否かを判断する。

【0101】

ステップS119において、左右の少なくとも片側に境界線があり、かつ、上下のいずれにも境界線がない状態ではないと判断された場合、注目画素は、左右のいずれにも境界線がなく、上下のいずれかに境界線があるので、ステップS120において、図22を用いて後述する画素値変換処理1を実行し、その後、処理はステップS122に進む。

【0102】

ステップS119において、左右の少なくとも片側に境界線があり、かつ、上下のいずれにも境界線がないと判断された場合、ステップS121において、画素値変換部36は、図23を用いて後述する画素値変換処理2を実行し、その後、処理はステップS122に進む。

【0103】

画素値変換部36は、ステップS122において、ステップS120もしくはステップS121において変換された画素値を出力し、ステップS123において、画素値変換用レジスタの値を $J=J+1$ とする。そして処理はステップS114に戻り、それ以降の処理が繰り返される。

【0104】

次に、図22のフローチャートを参照して、図21のステップS120における、画素値算出処理1について説明する。ここでは、BlendAおよびBlendBという値を用いて、画素値に対する重み付けを行う。

【0 1 0 5】

ステップ S 1 3 1 において、画素値変換部 3 6 は、(I, J) と (I, J + 1) の間に境界線が存在するか否かを判断する。

【0 1 0 6】

ステップ S 1 3 1 において、(I, J) と (I, J + 1) の間に境界線が存在しないと判断された場合、(I, J) と (I, J - 1) の間に境界線が存在しているということであり、処理は、ステップ S 1 3 4 に進む。ステップ S 1 3 1 において、(I, J) と (I, J + 1) の間に境界線が存在すると判断された場合、ステップ S 1 3 2 において、画素値変換部 3 6 は、図 1 9 を用いて説明した、距離検出処理の結果を参照し、境界線は、(I, J) のほうに近いかな否かを判断する。

【0 1 0 7】

ステップ S 1 3 2 において、境界線が (I, J) から遠いと判断された場合、処理は、ステップ S 1 3 4 に進む。ステップ S 1 3 2 において、境界線が (I, J) に近いと判断された場合、ステップ S 1 3 3 において、画素値変換部 3 6 は、 $BlendA = 2 \times ((I, J) \text{ から境界線までの距離} \div (\text{画素間の距離}))$  とする。

【0 1 0 8】

ステップ S 1 3 1 において、(I, J) と (I, J + 1) の間に境界線が存在しないと判断された場合、およびステップ S 1 3 2 において、境界線が (I, J) から遠いと判断された場合、ステップ S 1 3 4 において、画素値変換部 3 6 は、 $BlendA = 1$  とする。

【0 1 0 9】

ステップ S 1 3 5 において、画素値変換部 3 6 は、(I, J) と (I, J - 1) の間の境界線は、(I, J) に近いかな否かを判断する。ステップ S 1 3 5 において、境界線が (I, J) から遠いと判断された場合、ステップ S 1 3 6 において、画素値変換部 3 6 は、 $BlendB = 1$  とする。ステップ S 1 3 5 において、境界線が (I, J) に近いと判断された場合、ステップ S 1 3 7 において、画素値変換部 3 6 は、 $BlendB = 2 \times ((I, J) \text{ から境界線までの距離} \div (\text{画素間の距離}))$  とする。

【0 1 1 0】

ステップ S 1 3 8 において、画素値変換部 3 6 は、出力画像 = { [ 入力された ( I , J ) の画素値  $\times$  ( BlendA + BlendB ) ] + [ 入力された ( I , J - 1 ) の画素値  $\times$  ( 1 - BlendA ) ] + [ 入力された ( I , J + 1 ) の画素値  $\times$  ( 1 - BlendB ) ] }  $\div$  2 とし、処理は、図 2 1 のステップ S 1 2 2 に進む。

【0 1 1 1】

次に、図 2 3 のフローチャートを参照して、図 2 1 のステップ S 1 2 1 における、画素値算出処理 2 について説明する。ここでも、BlendA および BlendB という値を用いて、画素値に対する重み付けを行う。

【0 1 1 2】

ステップ S 1 4 1 において、画素値変換部 3 6 は、( I , J ) と ( I - 1 , J ) の間に境界線が存在するか否かを判断する。

【0 1 1 3】

ステップ S 1 4 1 において、( I , J ) と ( I - 1 , J ) の間に境界線が存在しないと判断された場合、( I , J ) と ( I + 1 , J ) の間に境界線が存在することであり、処理は、ステップ S 1 4 4 に進む。ステップ S 1 4 1 において、( I , J ) と ( I - 1 , J ) の間に境界線が存在すると判断された場合、ステップ S 1 4 2 において、画素値変換部 3 6 は、図 1 9 を用いて説明した、距離検出処理の結果を参照し、境界線は、( I , J ) のほうに近いかな否かを判断する。

【0 1 1 4】

ステップ S 1 4 2 において、境界線が ( I , J ) から遠いと判断された場合、処理は、ステップ S 1 4 4 に進む。ステップ S 1 4 2 において、境界線が ( I , J ) に近いと判断された場合、ステップ S 1 4 3 において、画素値変換部 3 6 は、 $\text{BlendA} = 2 \times ( ( I , J ) \text{ から境界線までの距離} \div ( \text{画素間の距離} )$  とする。

【0 1 1 5】

ステップ S 1 4 1 において、( I , J ) と ( I - 1 , J ) の間に境界線が存在しないと判断された場合、およびステップ S 1 4 2 において、境界線が ( I , J ) から遠いと判断された場合、ステップ S 1 4 4 において、画素値変換部 3 6 は

、BlendA=1 とする。

【0 1 1 6】

ステップ S 1 4 5 において、画素値変換部 3 6 は、(I, J) と (I + 1, J) の間の境界線は、(I, J) に近いかな否かを判断する。ステップ S 1 4 5 において、境界線が (I, J) から遠いと判断された場合、ステップ S 1 4 6 において、画素値変換部 3 6 は、BlendB=1 とする。ステップ S 1 4 5 において、境界線が (I, J) に近いと判断された場合、ステップ S 1 4 7 において、画素値変換部 3 6 は、BlendB=

$2 \times ((I, J) \text{ から境界線までの距離} \div (\text{画素間の距離}))$  とする。

【0 1 1 7】

ステップ S 1 3 8 において、画素値変換部 3 6 は、出力画像={ [入力された (I, J) の画素値  $\times$  (BlendA+BlendB)] + [入力された (I-1, J) の画素値  $\times$  (1-BlendA)] + [入力された (I+1, J) の画素値  $\times$  (1-BlendB)] }  $\div 2$  とし、処理は、図 2 1 のステップ S 1 2 2 に進む。

【0 1 1 8】

図 5 を用いて説明したように、注目する画素 (I, J) を中心とする、一辺が表示画像の画素間の距離  $\alpha$  と等しい正方形 4 の内部の画素値の平均値を求めて、その値を表示画像の画素値とすれば、ジャギーは除去されるのであるが、正方形 4 の内部の注目画素の位置に従った、白と黒との画素値の重み付けを求める処理が、図 2 3 のステップ S 1 4 3 およびステップ S 1 4 6、もしくは、ステップ S 1 4 4 およびステップ S 1 4 7 に対応する。そして、正方形 4 の内部の画素値の平均値を求める処理が、ステップ S 1 4 8 に対応する。なお、図 2 3 のフローチャートからもわかるように、画素値の変更を行うべきではない場合（例えば、境界線が画素間の中央を通過している場合）は、BlendA および BlendB の値は 1 となるため、ステップ S 1 4 8 において、実質的には画素値の変更は行われな

【0 1 1 9】

以上の処理により、入力された画素の画素値は、表示した際にジャギーが起こらないように変換され、出力される。この例においては、注目画素と、境界線との距離によって重み付けを行ったが、これ以外の要素によって重み付けを行うよ

うにしてもよい。

#### 【0 1 2 0】

なお、これらの処理は、例えば、縦方向の出力画素数を、入力画素数に対して 2 倍にすることにより、インターレース画像（奇数行のみ走査された画像）から、プログレッシブ画像（全ての行を走査された画像）を生成する場合に適応することもできる。

#### 【0 1 2 1】

出力画素の間隔は、図 2 4（A）に示されるように、横方向は入力画素と等しく、縦方向は入力画素の半分となる。プログレッシブ画像の出力のために、新たに作成される画素の画素値は、新たに作成される画像の近辺に境界線がない場合（図 2 4（A）の場合）、新たに作成される画素の真上の画素と真下の画素の画素値の平均値となる。

#### 【0 1 2 2】

入力画素の画素間の距離を  $\alpha$  として、新たに作成される画素に対して、縦方向に  $\alpha/4$  以下の距離、もしくは横方向に  $\alpha/2$  以下の距離内（すなわち、新たに作成される画素を中心とした、縦  $\alpha/2$ 、横  $\alpha$  の長方形の範囲内）に境界線がある場合について、図 2 4（B）を用いて説明する。

#### 【0 1 2 3】

例えば、図 2 4（B）のように、新たに作成される画素の近辺に、横方向の境界線がある場合、 $(I-1, J)$  と  $(I, J)$  との間に新たに作成される画素の画素値は、真上の画素  $(I-1, J)$  と真下の画素  $(I, J)$  の画素値に対して、境界線 1 と出力画素との距離に応じて重み付けして演算することによって求められる。すなわち、 $(I-1, J)$  と  $(I, J)$  との間に新たに作成される画素の画素値は、図中の  $\beta$  と  $\beta'$  の比で、白の画素と黒の画素が重み付けされる。新たに作成される画素の近辺に、縦方向の境界線がある場合においても同様である。

#### 【0 1 2 4】

このようにして求められるプログレッシブ画像の例を図 2 5 に示す。境界線近辺の画素は、画素と境界線の距離に応じた重み付けがなされたグレーの画素とし



て出力される。従って、プログレッシブ画像のジャギーを除去することができる。

#### 【0125】

上述した一連の処理を実行するソフトウェアは、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば汎用のパーソナルコンピュータなどに、記録媒体からインストールされる。

#### 【0126】

この記録媒体は、図6に示すように、コンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク21（フロッピーディスクを含む）、光ディスク22（CD-ROM（Compact Disk-Read Only Memory）、DVD（Digital Versatile Disk）を含む）、光磁気ディスク23（MD（Mini-Disk）を含む）、もしくは半導体メモリ24などよりなるパッケージメディアなどにより構成される。

#### 【0127】

また、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

#### 【0128】

##### 【発明の効果】

請求項1に記載の境界線検出装置、請求項4に記載の境界線検出方法、請求項5に記載の記録媒体によれば、入力された複数の画素を保存し、保存されている画素の画素値の差分を算出し、画素値の差分から、境界の有無を検出し、画素値の差分、および境界の有無の情報を基に、境界線の情報を作成するようにしたので、サンプリングレートを増やすことなく、簡単な構成で境界線を検出することができる。

#### 【0129】

請求項 6 に記載の画像処理装置、請求項 7 に記載の画像処理方法、請求項 8 に記載の記録媒体においては、入力された複数の画素を保存し、画素に隣接する境界線の有無を検出し、画素に対する境界線の位置を算出し、算出された位置に従って、画素の画素値に対して重み付けを実行し、重み付けが施された画素を出力するようにしたので、サンプリングレートを増やすことなく、より簡単な構成でジャギーを除去することができる。

【図面の簡単な説明】

【図 1】

浮動小数点演算により計算された画像の例を示す図である。

【図 2】

各画素と境界線の位置関係を説明するための図である。

【図 3】

ジャギーを説明するための図である。

【図 4】

ジャギーを除去するために好ましい出力画素を説明するための図である。

【図 5】

ボックスフィルタを説明するための図である。

【図 6】

画像処理装置の構成を示すブロック図である。

【図 7】

画像処理部の構成を示すブロック図である。

【図 8】

注目画素に対する境界線検出行および境界線情報を説明するための図である。

【図 9】

境界線検出処理を説明するためのフローチャートである。

【図 1 0】

横境界線特定処理を説明するためのフローチャートである。

【図 1 1】

横境界線特定処理を説明するためのフローチャートである。

【図 1 2】

注目画素と境界線の関係の説明するための図である。

【図 1 3】

注目画素と境界線の関係の説明するための図である。

【図 1 4】

注目画素と境界線の関係の説明するための図である。

【図 1 5】

注目画素と境界線の関係の説明するための図である。

【図 1 6】

注目画素と境界線の関係の説明するための図である。

【図 1 7】

縦境界線特定処理について説明するためのフローチャートである。

【図 1 8】

注目画素と境界線の関係の説明するための図である。

【図 1 9】

距離検出処理について説明するためのフローチャートである。

【図 2 0】

距離算出の条件と計算式を説明するための図である。

【図 2 1】

境界線検出処理について説明するためのフローチャートである。

【図 2 2】

図 2 1 の画素値算出処理 1 について説明するためのフローチャートである。

【図 2 3】

図 2 1 の画素値算出処理 2 について説明するためのフローチャートである。

【図 2 4】

インターレース画像からプログレッシブ画像を生成する処理について説明するための図である。

【図 2 5】

インターレース画像からプログレッシブ画像を生成する処理について説明する

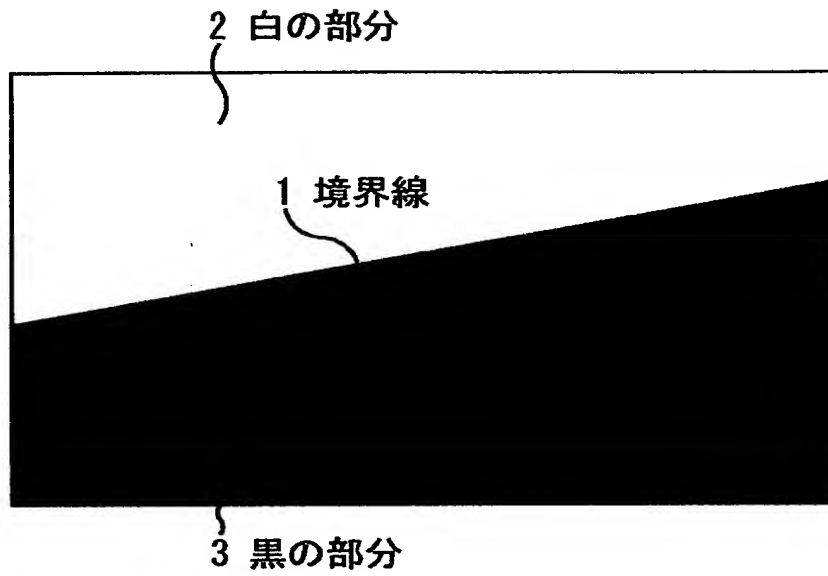
ための図である。

【符号の説明】

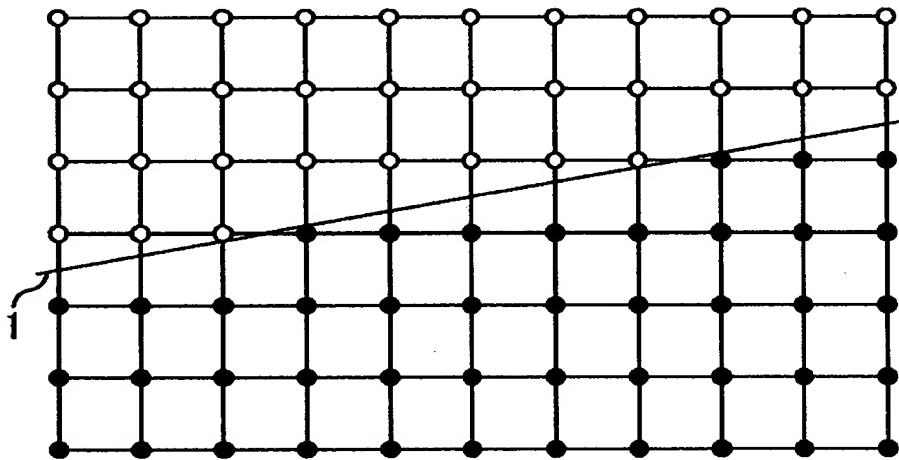
1 1 制御部, 1 8 モニタ, 2 5 画像処理部, 3 1 メモリ, 3  
2 横方向境界線検出部, 3 3 横方向境界線情報メモリ, 3 4 縦方向境  
界線検出部, 3 5 縦方向境界線情報メモリ, 3 6 画素値変換部

【書類名】図面

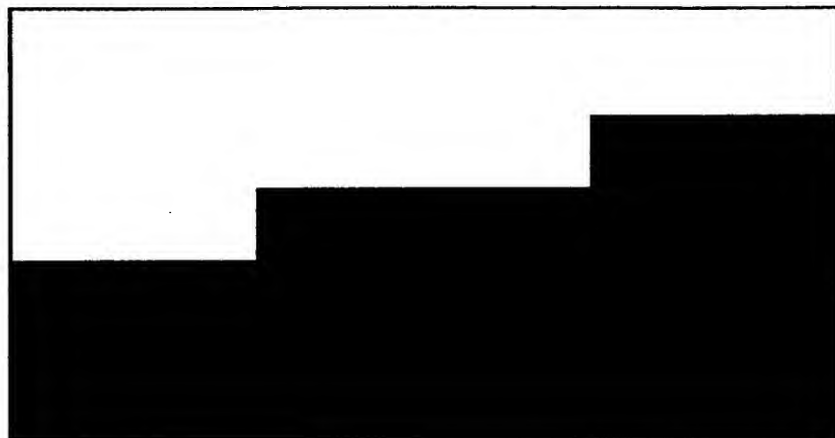
【図 1】



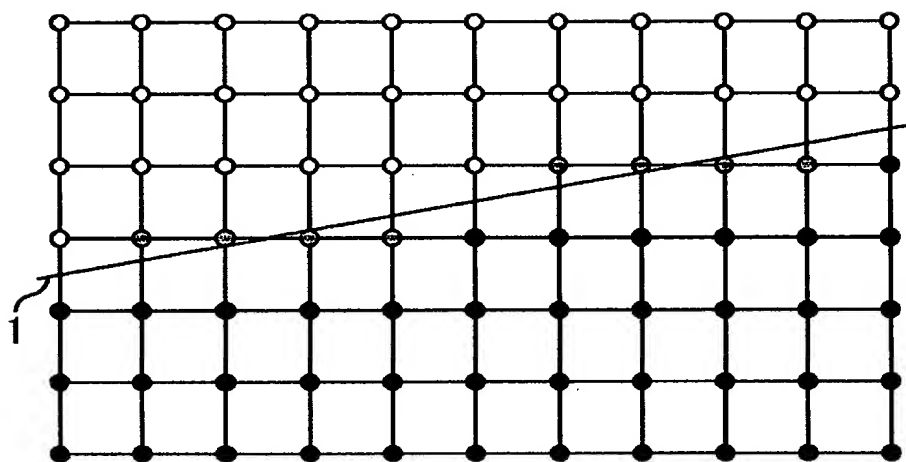
【図 2】



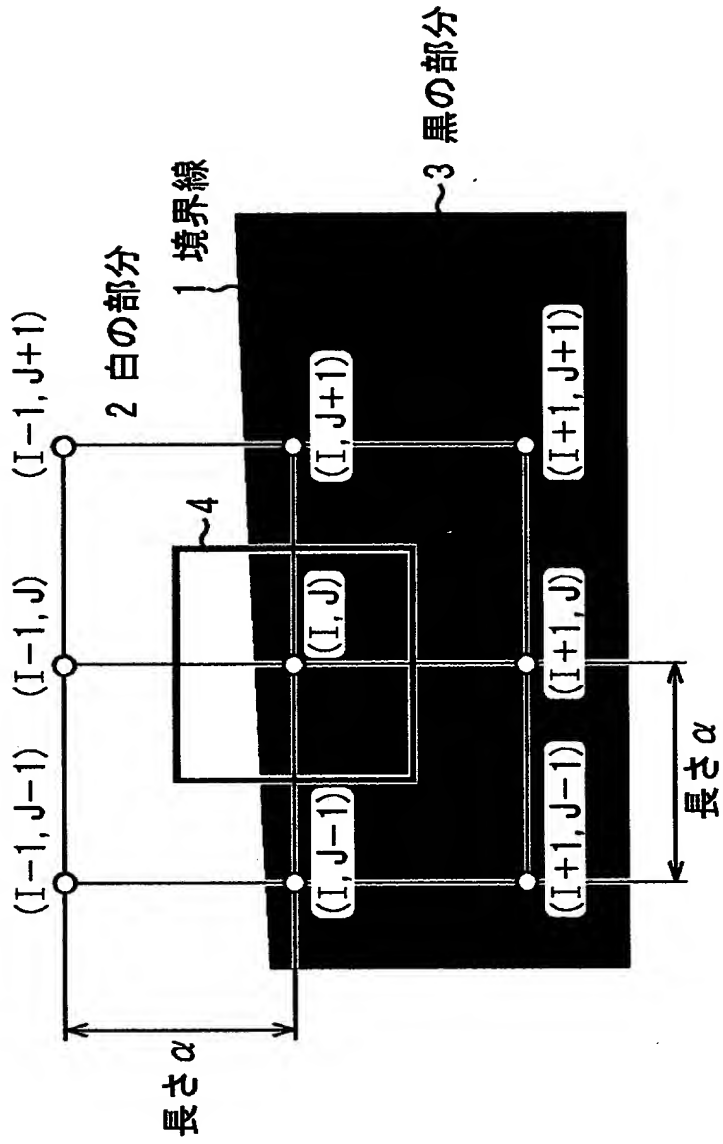
【図 3】



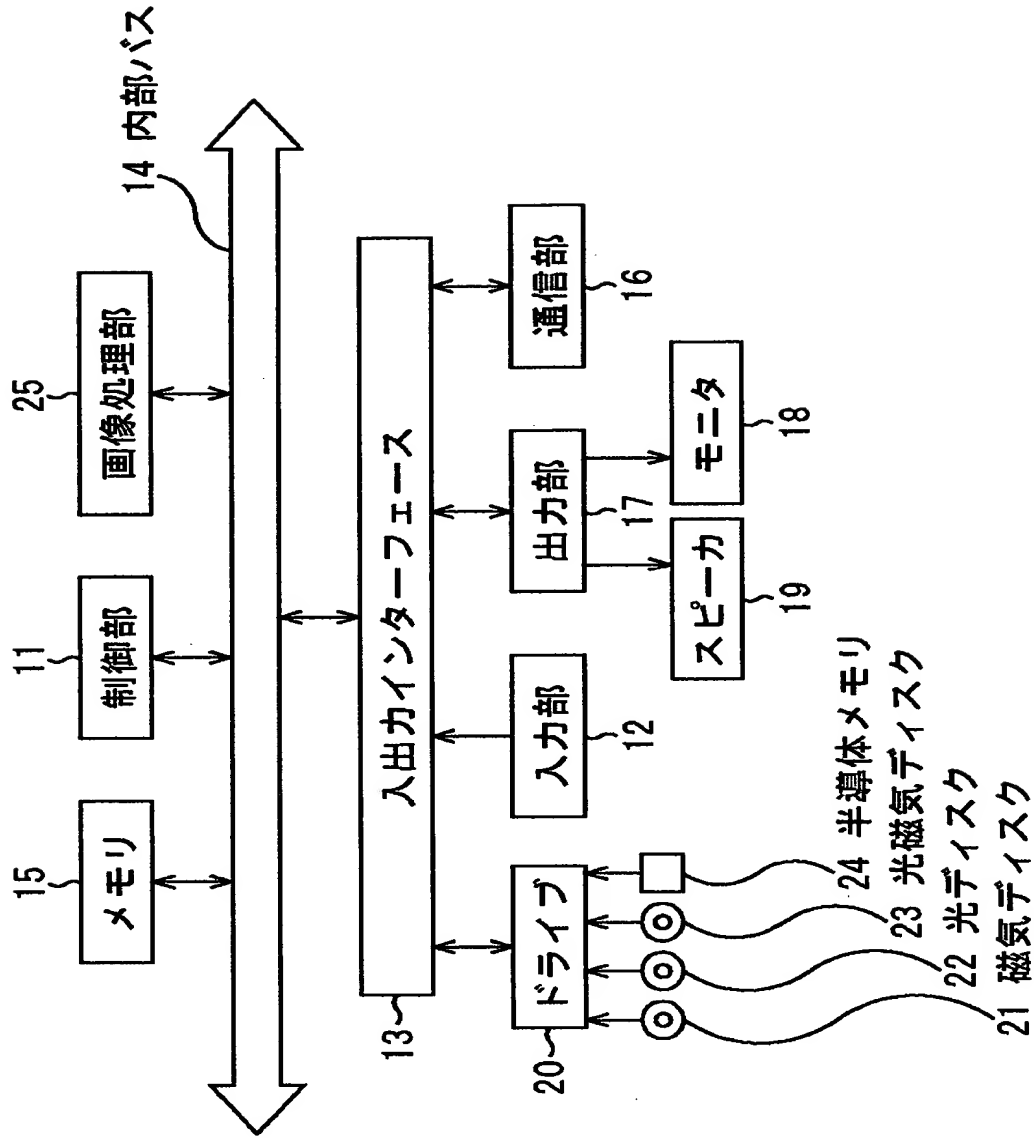
【図 4】



【図 5】

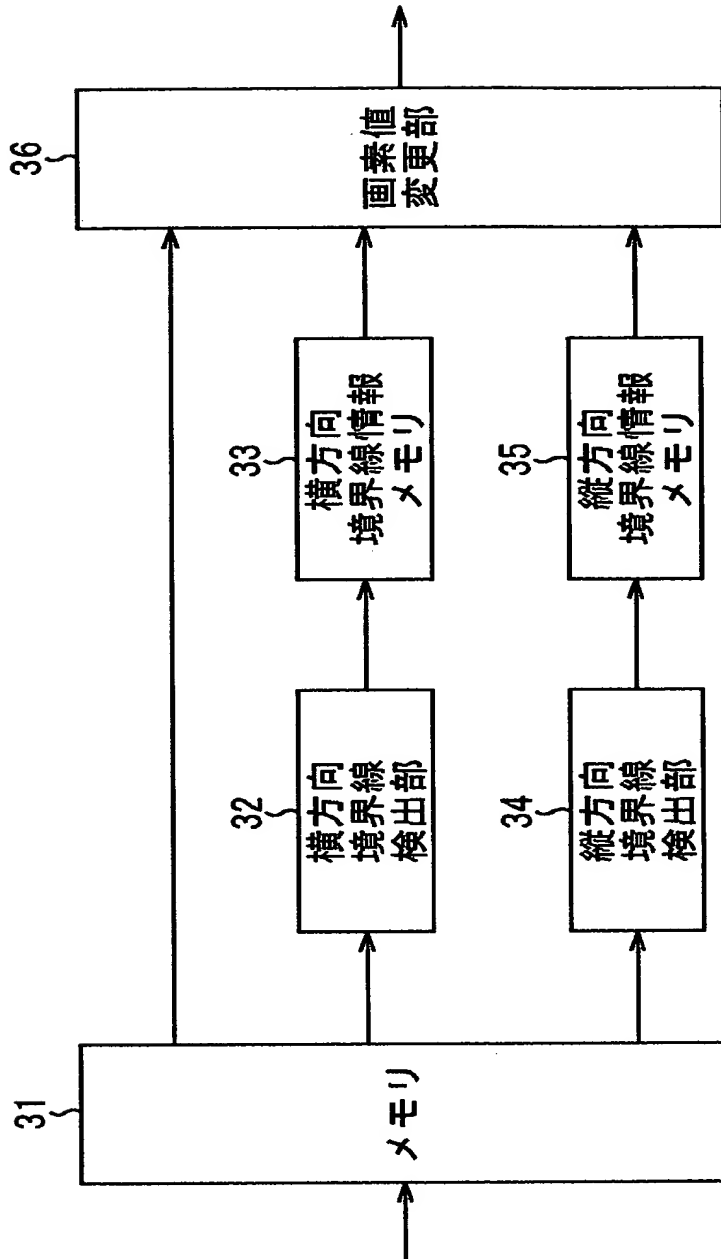


【図 6】



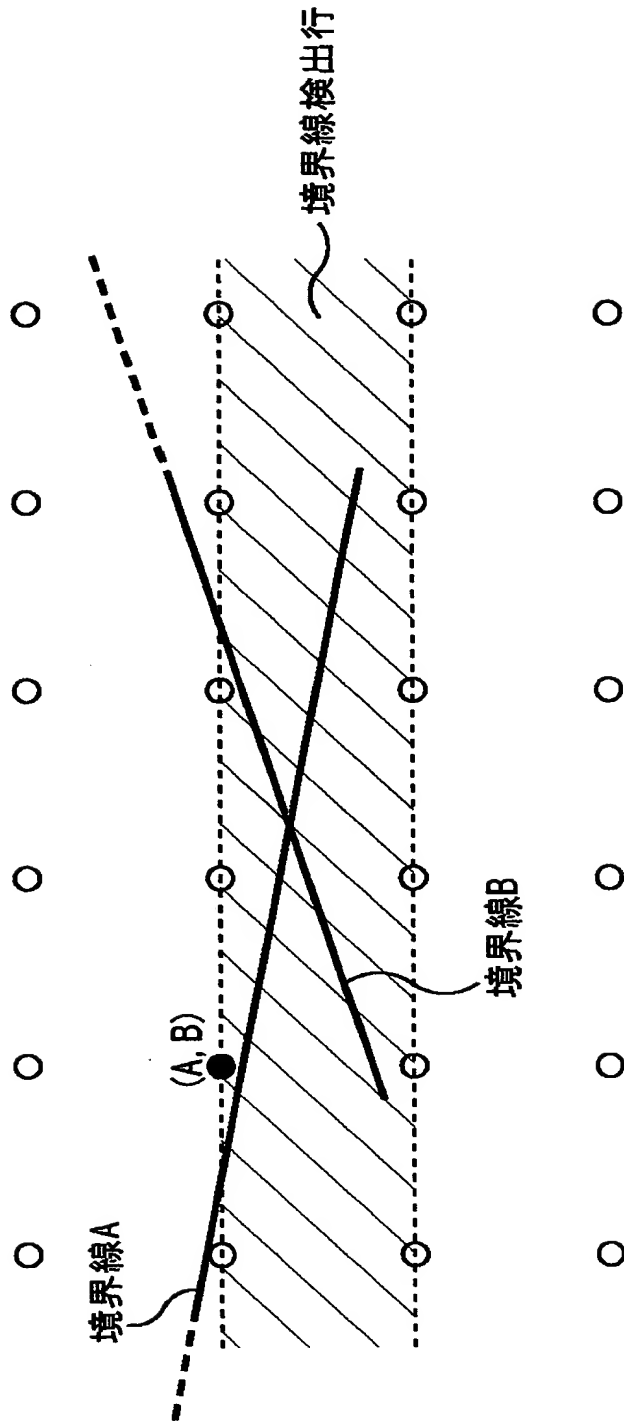


【図 7】

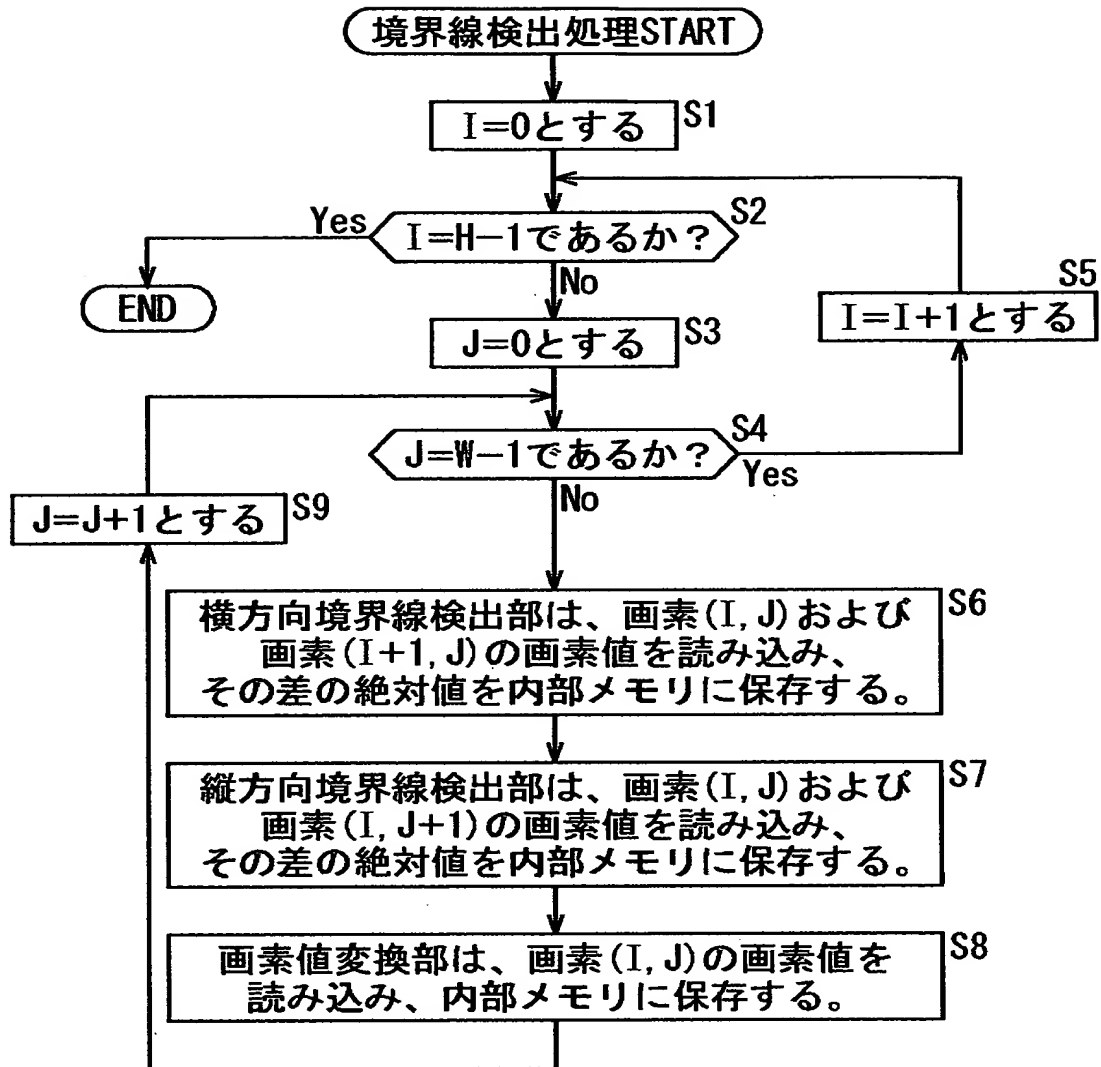


画像処理部 25

【图 8】

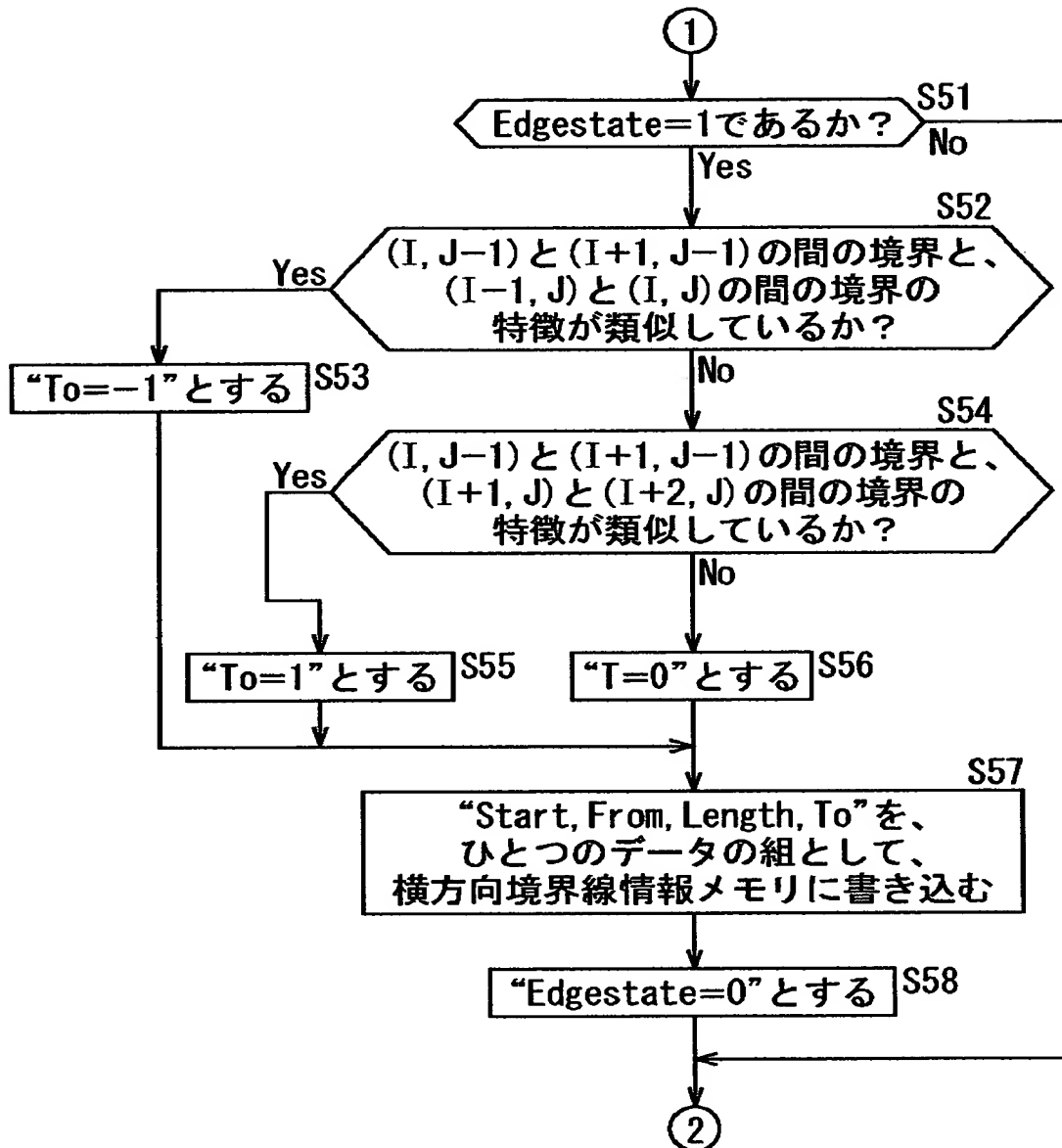


【図 9】

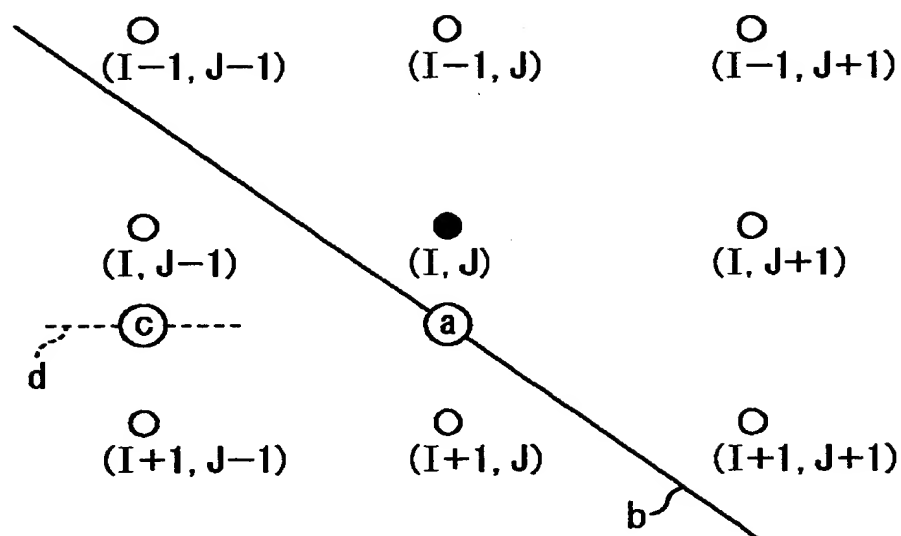




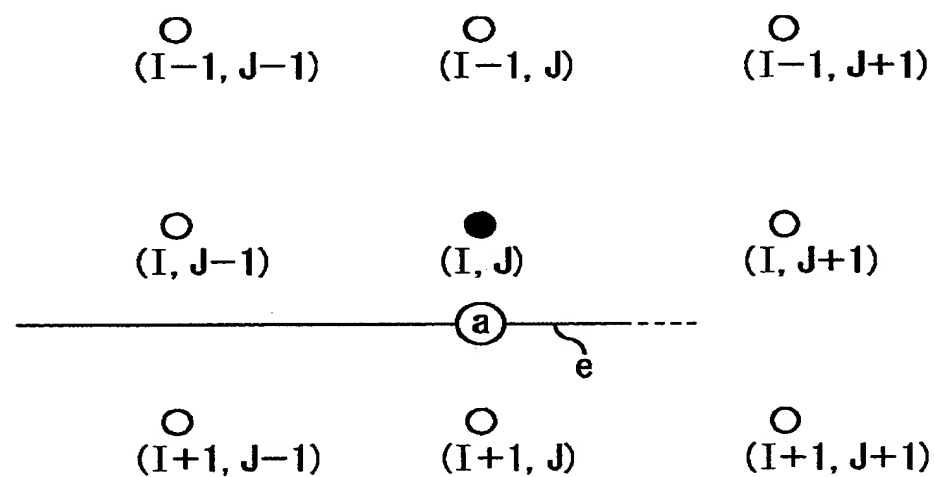
【図 1 1】



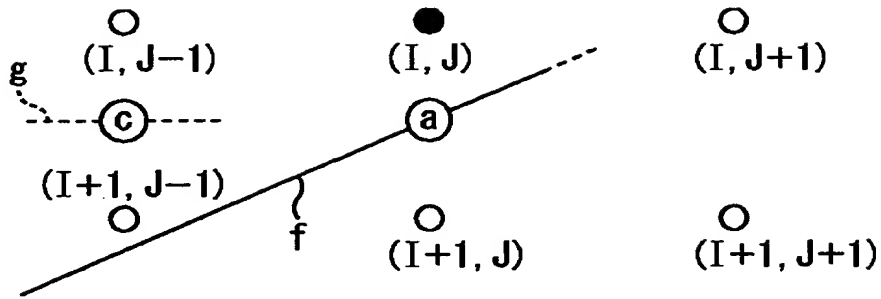
【図 1 2】



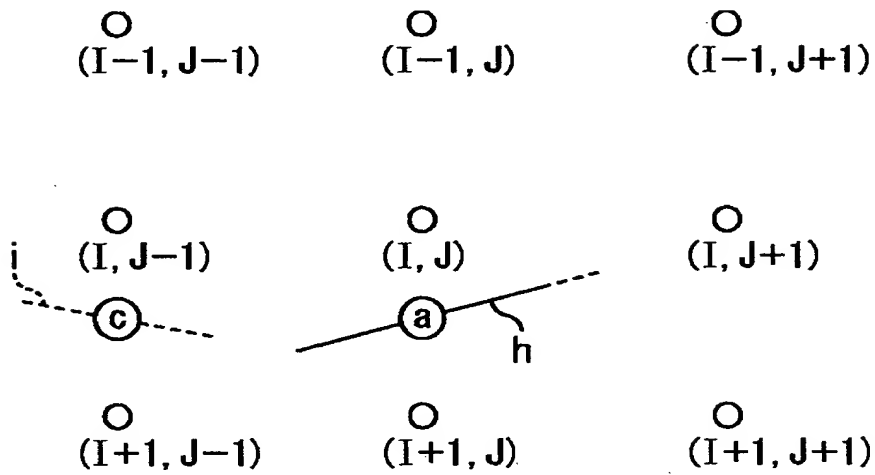
【図 1 3】



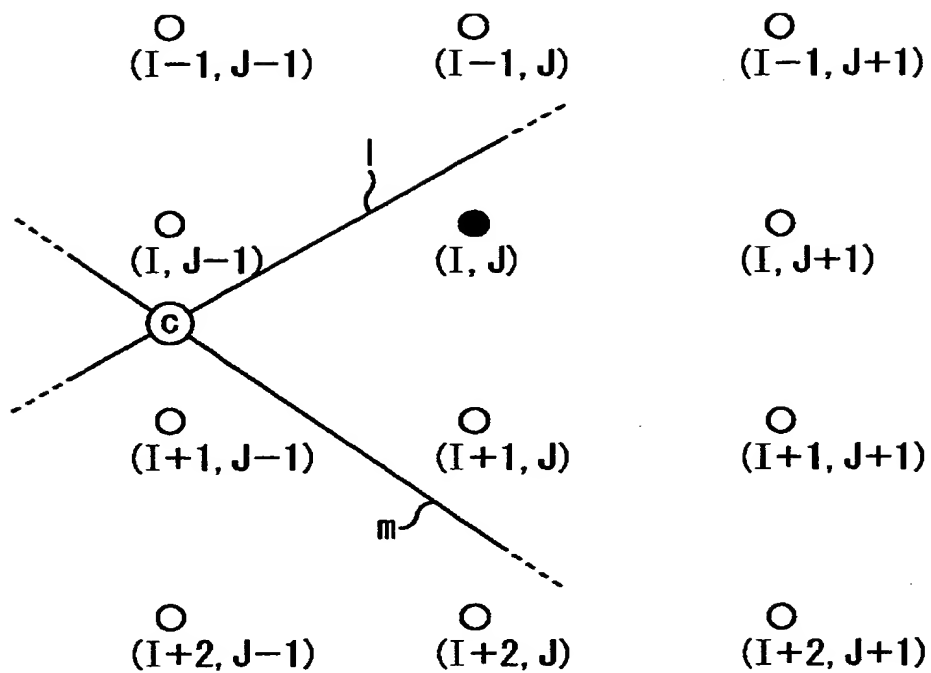
【图 1 4】



【图 1 5】

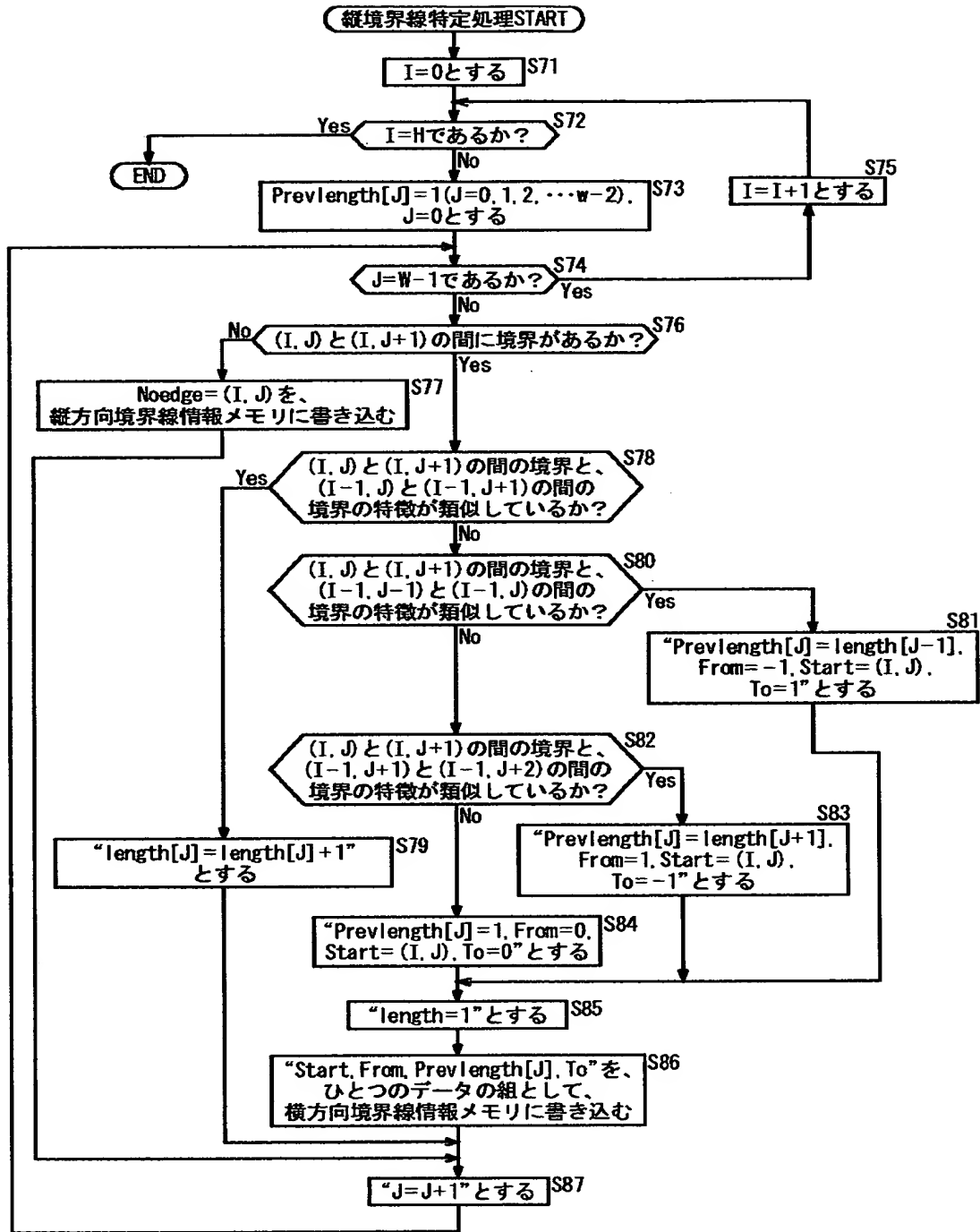


【图 1 6】

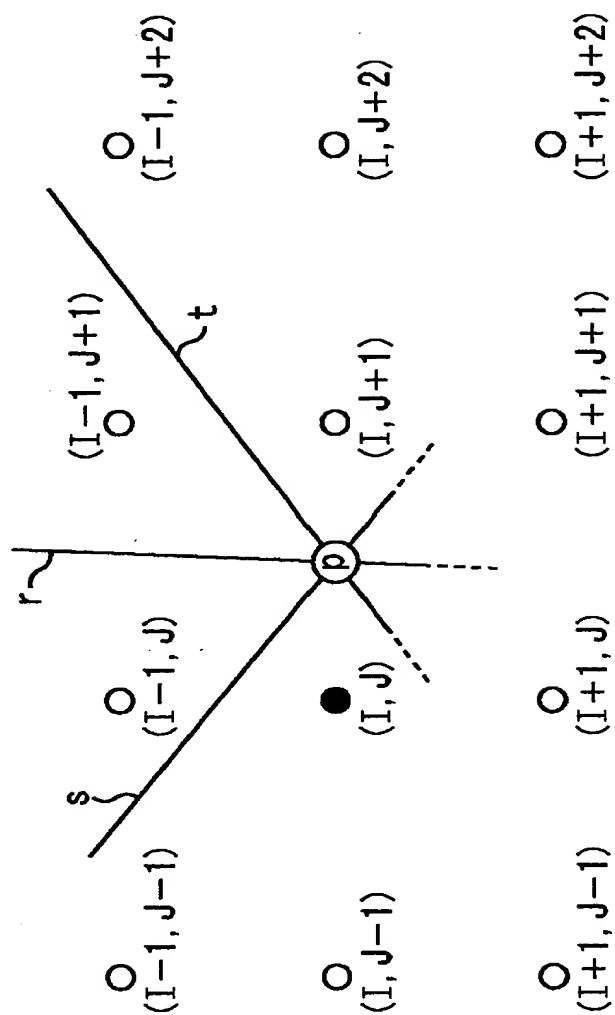




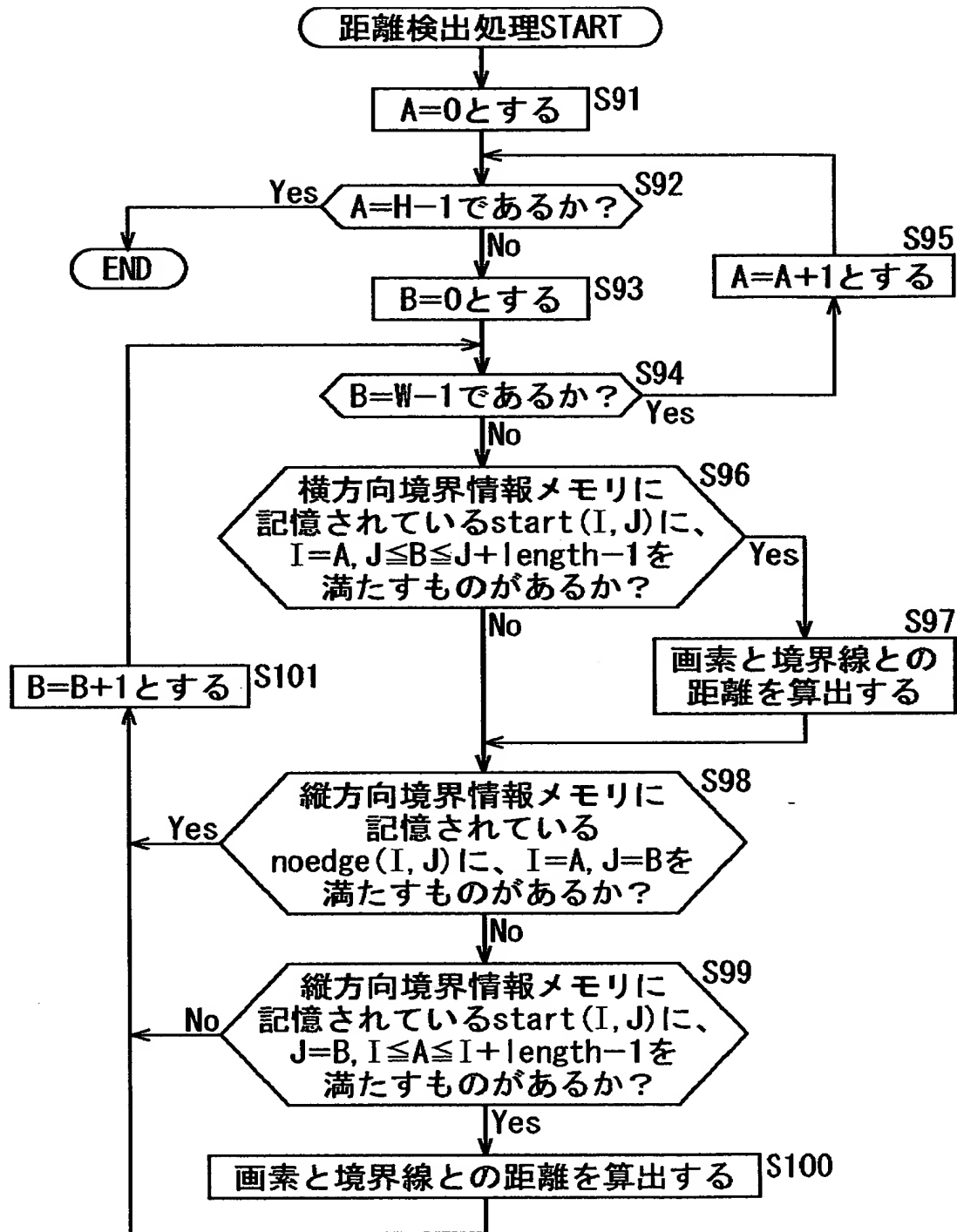
【図 1 7】



【図 1 8】



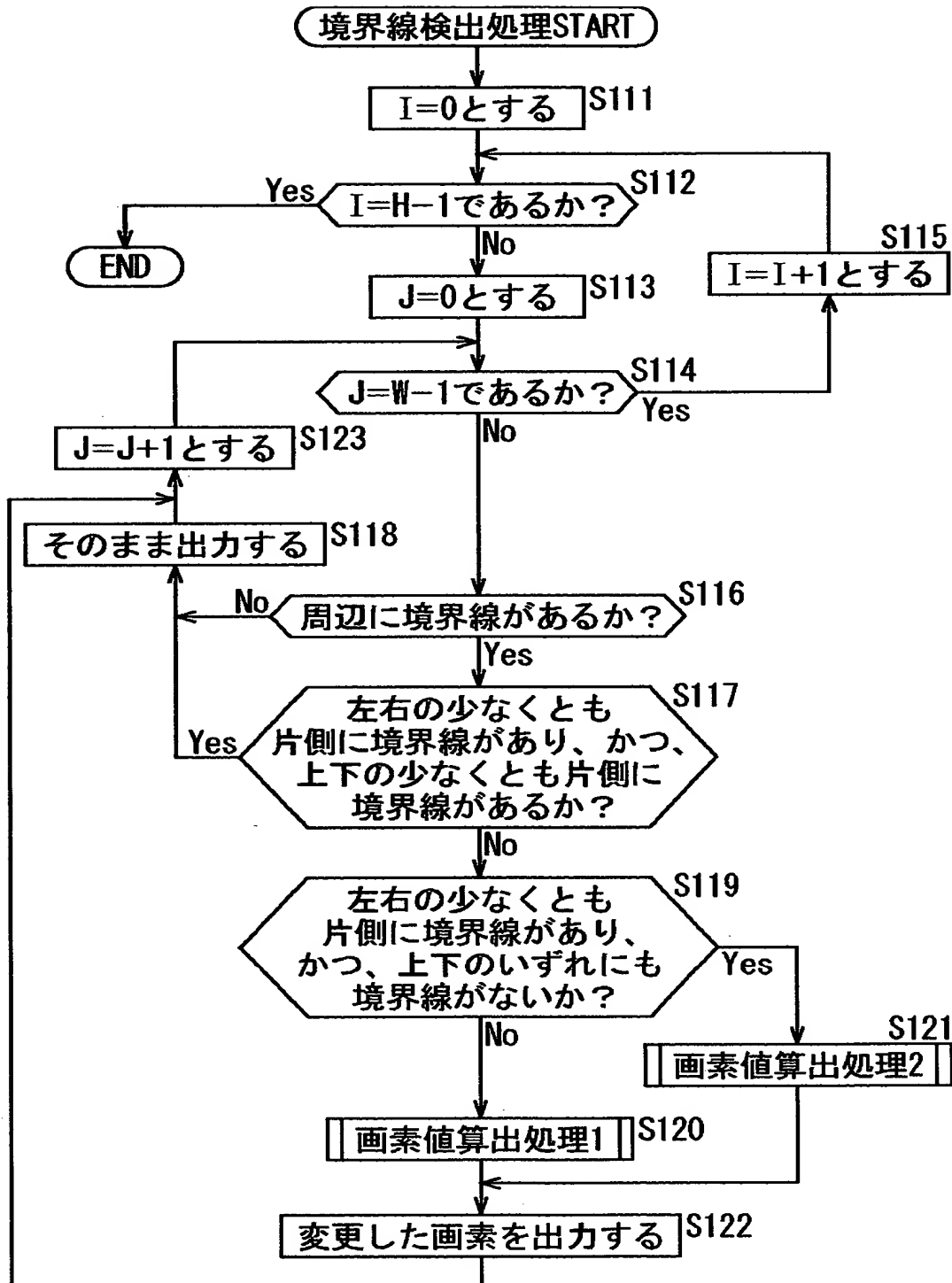
【図 1 9】



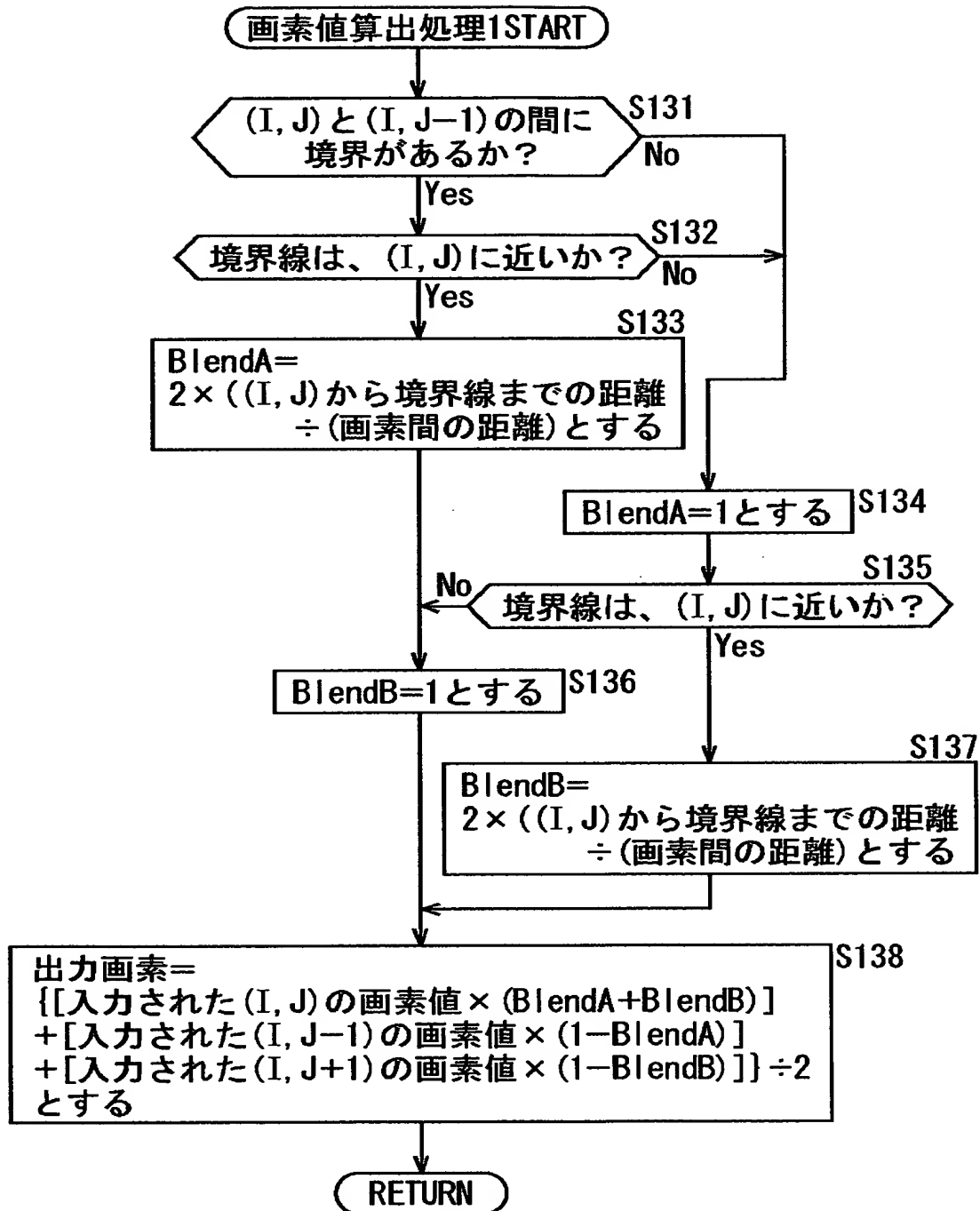
【図 2 0】

Fronの値	Toの値	Bの値	(A, B) 側からの距離
1	1	$B < j + (\text{length}/2) \cdot 0.5$	$1 - (1 \div \text{length} \times (B - j + 0.5))$
1	1	$B > j + (\text{length}/2) \cdot 0.5$	$1 \div \text{length} \times (B - j + 0.5)$
1	0	-	$1 - (0.5 \div \text{length} \times (B - j + 0.5))$
1	-1	-	$1 - (1 \div \text{length} \times (B - j + 0.5))$
0	1	-	$0.5 + (0.5 \div \text{length} \times (B - j + 0.5))$
0	0	-	0
0	-1	-	$0.5 - (0.5 \div \text{length} \times (B - j + 0.5))$
-1	1	-	$1 \div \text{length} \times (B - j + 0.5)$
-1	0	-	$0.5 \div \text{length} \times (B - j + 0.5)$
-1	-1	$B < j + (\text{length}/2) \cdot 0.5$	$1 \div \text{length} \times (B - j + 0.5)$
-1	-1	$B > j + (\text{length}/2) \cdot 0.5$	$1 - (1 \div \text{length} \times (B - j + 0.5))$

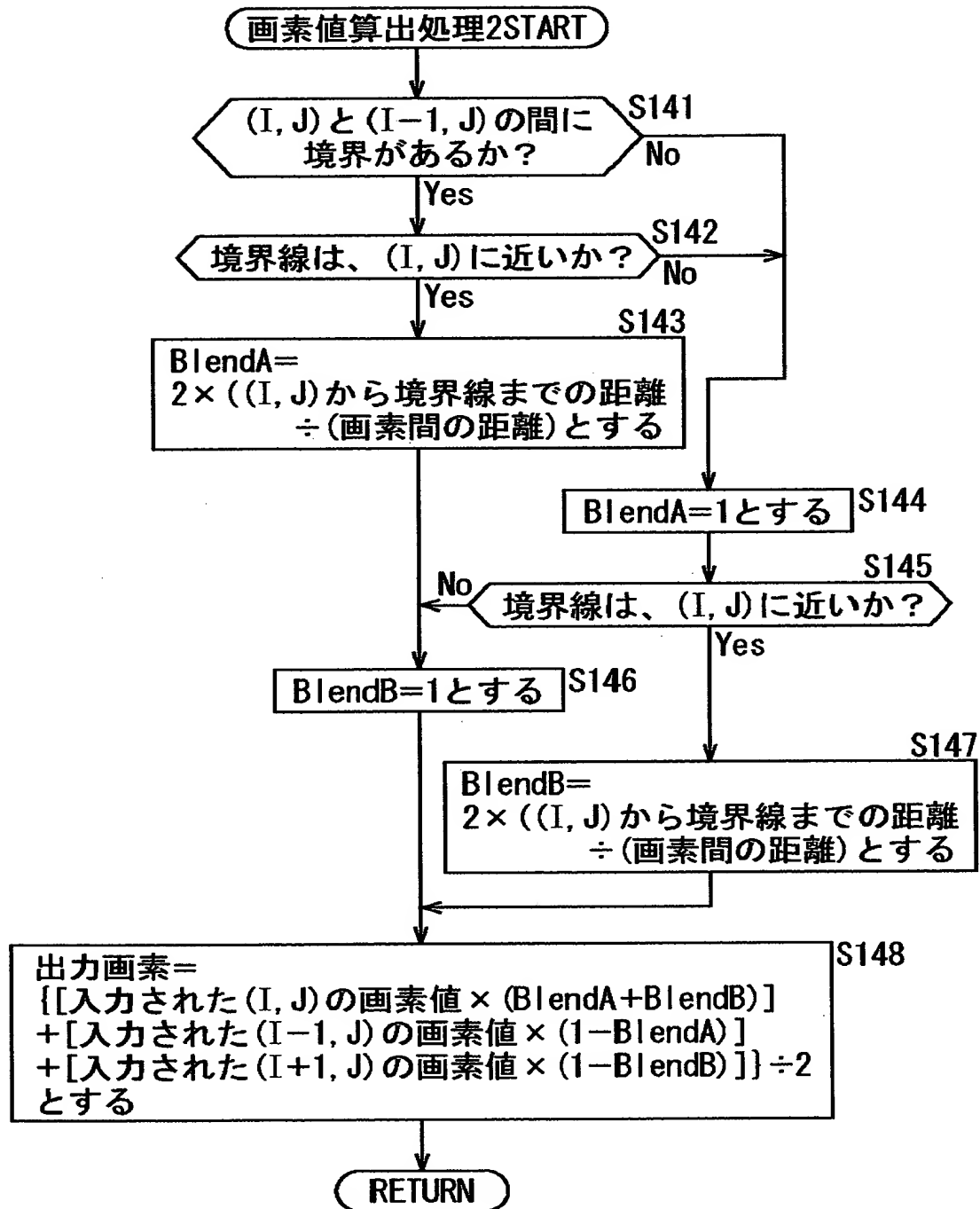
【図 2 1】



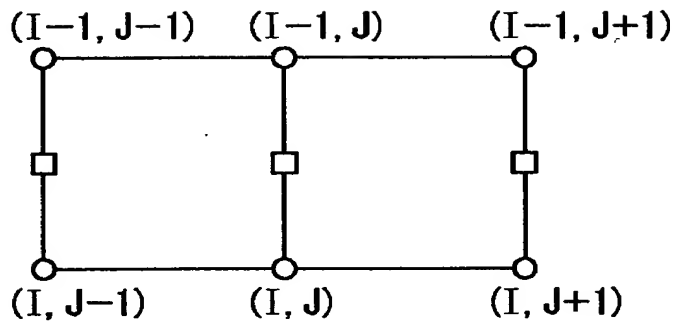
【図 2 2】



【図 2 3】

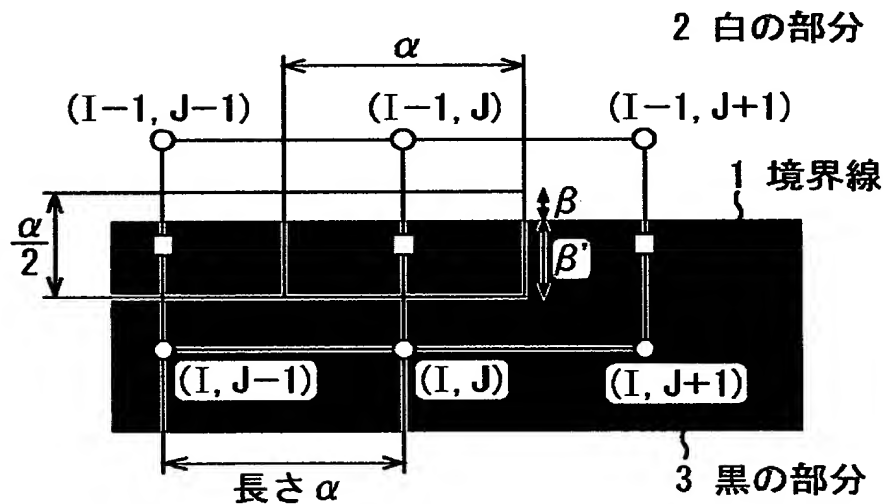


【図 2 4】



○ = 入力画像の画素  
□ = 出力画像の為に  
新しく作成する画素

(A)

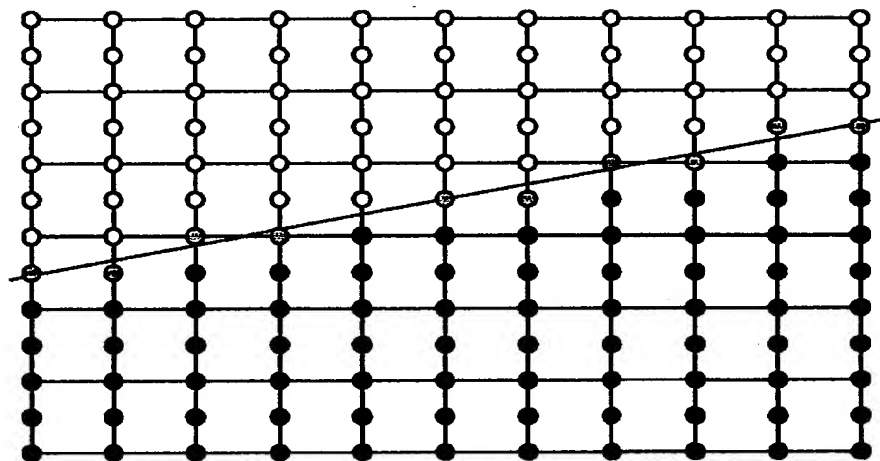


○ = 入力画像の画素  
□ = 出力画像の為に  
新しく作成する画素

(B)



【図 2 5】



【書類名】 要約書

【要約】

【課題】 サンプリングレートを増やすことなく、境界線付近のジャギーを除去する。

【解決手段】 注目画素（A，B）と、それに隣接する画素との画素値の差を比較することにより、注目画素の近辺に境界が存在するか否かを検出し、その境界線の長さ、方向、および始点と終点を求める。そして、注目画素と境界線との距離を算出することにより、注目画素の画素値に対する重み付けを行い、重み付けを施した画素を出力する。

【選択図】 図 8

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日	1990年 8月30日
[変更理由]	新規登録
住 所	東京都品川区北品川6丁目7番35号
氏 名	ソニー株式会社